

Noël-Arnaud MAGUIS

RÉDIGEZ DES DOCUMENTS DE QUALITÉ AVEC

L^AT_EX

L'OUTIL DES PROFESSIONNELS POUR PUBLIER MÉMOIRES,
THÈSES, RAPPORTS, ARTICLES SCIENTIFIQUES...



Issu du célèbre
Site du Zéro
www.siteduzero.com



www.siteduzero.com

Noël-Arnaud MAGUIS

RÉDIGEZ DES DOCUMENTS DE QUALITÉ AVEC

L^AT_EX

**L'OUTIL DES PROFESSIONNELS POUR PUBLIER MÉMOIRES,
THÈSES, RAPPORTS, ARTICLES SCIENTIFIQUES...**



www.siteduzero.com

DANS LA MÊME COLLECTION



APPRENEZ À PROGRAMMER EN C

MATHIEU NEBRA
ISBN : 978-2-9535278-0-3



**CONCEVEZ VOTRE SITE WEB
AVEC PHP ET MYSQL**
MATHIEU NEBRA
ISBN : 978-2-9535278-1-0



**REPRENEZ LE CONTRÔLE À L'AIDE
DE LINUX**
MATHIEU NEBRA
ISBN : 978-2-9535278-2-7

Noël-Arnaud MAGUIS

RÉDIGEZ DES DOCUMENTS DE QUALITÉ AVEC

L^AT_EX

**L'OUTIL DES PROFESSIONNELS POUR PUBLIER MÉMOIRES,
THÈSES, RAPPORTS, ARTICLES SCIENTIFIQUES...**



www.siteduzero.com

zCorrecteurs.fr



Cet ouvrage a bénéficié des relectures attentives des zCorrecteurs.



Sauf mention contraire, le contenu de cet ouvrage est publié sous la licence :
Creative Commons BY-NC-SA 2.0

La copie de cet ouvrage est autorisée sous réserve du respect des conditions de la licence
Texte complet de la licence disponible sur : <http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

Simple IT 2010 - ISBN : 978-2-9535278-4-1

Avant-propos

L'histoire des sciences, et plus particulièrement celle de l'informatique, compte une foule d'inventions ayant connu un succès phénoménal, alors qu'elles n'étaient initialement destinées qu'à répondre à un besoin très localisé et identifié. Les exemples sont légion, et les citer ici prendrait des dizaines de pages. Toutefois, l'un d'eux vous a poussé à ouvrir ce livre : $\text{\LaTeX}$.

$\text{\LaTeX}$ est un langage créé par des scientifiques qui, lassés de ne publier leurs écrits qu'avec des mises en page disgracieuses, ont décidé d'écrire un programme capable de mettre en page à peu près n'importe quoi (que le document soit scientifique ou non), de gérer la mise en forme d'expressions mathématiques et d'offrir la possibilité à qui le veut d'ajouter des fonctionnalités.

À l'époque, ce projet était un OVNI : ce n'est que depuis quelques années qu'il est possible de se rendre compte des avancées permises par $\text{\LaTeX}$ dans le domaine de la communication scientifique. À l'origine un simple projet de recherche, ce langage est devenu une référence dans le monde scientifique et éducatif. Vous seriez surpris de connaître le nombre d'enseignants rédigeant leurs cours dans ce langage notamment grâce à la maniabilité qu'il leur offre !

Malgré cela, bien qu'incontournable et très apprécié dans le monde scientifique, les universités ne font que très rarement l'effort de mettre en place des modules de cours dédiés à l'apprentissage de $\text{\LaTeX}$. De même, il existe peu d'ouvrages pour débutants permettant aux particuliers d'apprendre simplement ce langage¹.

Ce livre tente de pallier ce manque en présentant $\text{\LaTeX}$ et en fournissant un cours simple, adapté à tous, du curieux au scientifique chevronné. Il a été écrit en collaboration avec des chercheurs, dans le but de transmettre le plus simplement possible les notions essentielles de $\text{\LaTeX}$ pour vous permettre la rédaction d'articles, de mémoires, de rapports de stage, de thèses et bien d'autres écrits. Toutes les notions seront développées à partir de zéro : les seuls prérequis sont de savoir allumer un ordinateur et de disposer d'une connexion à l'internet.

Sachez que j'ai écrit le cours grâce auquel j'aurais souhaité apprendre $\text{\LaTeX}$. Je souhaite qu'il vous guide dans vos premiers pas avec $\text{\LaTeX}$. Je vous invite à me suivre dans la découverte d'un langage qui deviendra, au fil des pages, le vôtre. . .

1. Il est loin d'être réservé aux professionnels. Sa simplicité le rend accessible à tous.

Qu'allez-vous apprendre en lisant ce livre ?

Le plan de ce livre a été conçu pour permettre à n'importe qui d'appréhender $\text{\LaTeX}$ sans difficulté. Voici le chemin que nous allons parcourir ensemble.

1. **Découverte de $\text{\LaTeX}$** : cette première partie sert à vous familiariser avec le langage, au niveau de son histoire, de ses fonctionnalités ou de ses petites particularités. Elle démarre par un chapitre introductif développant les spécificités de $\text{\LaTeX}$ et ses différences par rapport à Word. Tous les concepts essentiels y seront mentionnés pour vous permettre de travailler sereinement par la suite. C'est aussi dans cette partie que nous installerons les logiciels nécessaires à l'apprentissage du langage.
2. **Utilisation basique de $\text{\LaTeX}$** : cette partie sera notre premier contact concret avec $\text{\LaTeX}$. Il n'y a aucune difficulté dans les quatre chapitres qui la composent. Cependant, à la fin de ceux-ci, vous saurez modifier des marges, créer des listes, manipuler des citations et changer les polices dans vos documents. Un TP viendra clore cette partie pour vous permettre de mettre vos connaissances à l'épreuve et de créer un petit livre (ayant pour thème la Déclaration de droits de l'homme) entièrement en $\text{\LaTeX}$.
3. **Utilisation avancée de $\text{\LaTeX}$** : les concepts abordés ici sont un peu plus complexes, mais en valent la chandelle. À la fin de cette partie, vous saurez écrire des expressions mathématiques, manipuler des images et des tableaux. Les tables des matières, les bibliographies et l'index seront abordés, puis ce cours se terminera sur un dernier TP bien complet, vous faisant mettre en page le théorème de Pythagore et quelques expressions mathématiques.

De plus, cinq annexes ont été composées pour vous, elles vous aideront à travailler au quotidien avec $\text{\LaTeX}$.

Comment lire ce livre ?

Suivez l'ordre des chapitres

Lisez ce livre comme on lit un roman. Il a été conçu pour cela. Contrairement à beaucoup de livres techniques où il est courant de lire en diagonale et de sauter certains chapitres, il est ici très fortement recommandé de suivre l'ordre du cours, à moins que vous ne soyez déjà un peu expérimenté.

Pratiquez en même temps

Pratiquez régulièrement. Je le répéterai très souvent au fil du cours, car $\text{\LaTeX}$ est un langage qui malgré sa simplicité apparente doit être pratiqué afin d'acquérir des automatismes.

Utilisez les codes web !

Le Site du Zéro, dont est issu ce livre, vous propose ce que nous appellerons des « codes web ». Ce sont des codes à six chiffres, à entrer dans un formulaire prévu à cet effet sur une page du Site du Zéro, qui vous permettront d'être automatiquement redirigé vers un site web, sans que vous ayez à en recopier l'adresse.

Pour utiliser les codes web, rendez-vous sur la page suivante² :

<http://www.siteduzero.com/codeweb.html>

Un formulaire vous invitera à entrer votre code web. Faites un premier essai avec le code ci-dessous :

▷ Code web : 123456

Ces codes web ont deux utilités :

- vous faire télécharger les codes source présents dans ce livre, ce qui vous évitera d'avoir à les recopier lorsqu'ils sont un peu longs ;
- vous rediriger vers les sites web présentés tout au long du cours.

Ce système de redirection permet de tenir à jour le livre que vous avez entre les mains, sans que vous ayez besoin de le racheter systématiquement à chaque nouvelle édition. Si un site web change d'adresse, le code web à utiliser, lui, ne changera pas. Si un site web disparaît, le code web associé vous redirigera vers une page du Site du Zéro expliquant ce qu'il s'est passé et vous proposant une solution de substitution.

En somme, c'est un moyen d'assurer la pérennité de cet ouvrage sans que vous ayez à faire quoi que ce soit !

Ce livre est issu du Site du Zéro

Cet ouvrage reprend le cours L^AT_EX du Site du Zéro dans une édition revue, corrigée et augmentée de nouvelles annexes ainsi que de notes de bas de page.

Il reprend les éléments qui ont fait le succès des cours du site, c'est-à-dire leur approche progressive et pédagogique, le ton léger, les explications simples et accessibles, ainsi que des TP vous permettant de réellement pratiquer de façon autonome.

Ce livre est aussi une innovation, car il est le premier de la collection « Livre du Zéro » à ne pas avoir été écrit par Mathieu Nebra lui-même (le fondateur du Site du Zéro), mais par un membre de la communauté du site. C'est un grand honneur pour moi d'être le premier membre dont le cours est publié en livre. D'autres membres travaillent déjà sur de prochains livres.

2. Vous pouvez aussi utiliser le formulaire de recherche du Site du Zéro, section « Code Web ».

Un livre écrit en partenariat avec une université !

Ce livre a été écrit en partie au sein de l'IMB (Institut de Mathématiques de Bordeaux), situé sur le campus de l'Université Bordeaux 1. Cette faculté a la particularité d'être, depuis quelques années, classée par l'Université de Shanghai comme faisant partie de ce qu'il y a de mieux dans le domaine de l'informatique³ au niveau mondial.

Des chercheurs de diverses UFR⁴ en physique et chimie ont participé à la création de ce livre. De plus, les logiciels utilisés dans ce cours ont été choisis en fonction des usages et habitudes des étudiants de cette université.

Remerciements

Un livre de ce genre demande beaucoup d'énergie et de temps pour être produit. J'aimerais remercier ici tous ceux qui m'ont aidé et ont donné de leur temps pour que mon projet soit mené à bien.

- Malika TALBI, à qui je dois les illustrations de chaque chapitre. Elle est une partenaire privilégiée et une amie très précieuse.
- Adrien LE MERCIER, Xavier VAISSEAU, Sébastien DUBOS, Sarah FECHTENBAUM, Marion DUSARD, Cécile QUÉRÉ, Maxime LECLERCQ, Diane MARTAIRE et tous ceux qui parfois se sont couchés à trois heures du matin juste pour m'aider à corriger ou tester des bouts de code. Sans eux, il n'y aurait point eu de livre.
- Mathieu NEBRA (M@teo21) et Pierre DUBUC (karamilo), qui ont cru en mon projet et m'ont aidé tout au long du processus d'écriture.
- G. GODINAUD, E. ABRAHAM et C. ZAKRI qui ont soutenu mon initiative. Ils sont tous les trois chercheurs à Bordeaux 1, respectivement en math / info, physique et chimie.
- L'équipe des zCorrecteurs, une bande de grands acharnés de la langue française. Souvent plus jeunes que moi, ils ont su me surprendre par leur maîtrise de la typographie et de l'orthographe. Je tiens tout particulièrement à remercier Agnès HAASSER (Tûtie), Philippe LUTUN (ptipilou), Damien SMEETS (Karl Yeurl), Guillaume GAULLIER (Guillawme), Maxence CORDIEZ (Ziame) et Christophe TAFANI-DEREPPER (christophetd). Ils ont relu, corrigé l'intégralité du livre et se sont montrés particulièrement sympathiques avec moi.
- Poulpy et Monsieur Poule, les mascottes de ce livre, que vous découvrirez dans le chapitre 10.
- Fan Jiyong pour la couverture du livre.
- L'équipe de Simple IT et la communauté du Site du Zéro, qui m'ont permis de donner corps à l'un de mes rêves : signer un livre.

3. Bordeaux 1 ne compte pas qu'un secteur informatique, elle couvre une très grande partie des domaines scientifiques.

4. Unités de Formation et de Recherche.

Sommaire

Avant-propos	i
Qu’allez-vous apprendre en lisant ce livre?	ii
Comment lire ce livre?	ii
Ce livre est issu du Site du Zéro	iii
Un livre écrit en partenariat avec une université!	iv
Remerciements	iv
 I Découverte de L^AT_EX	 1
 1 Qu’est-ce que L^AT_EX ?	 3
L ^A T _E X, quésaco?	4
Word versus L ^A T _E X	5
L ^A T _E X et les publications	8
Historique rapide	10
 2 Installer L^AT_EX	 13
De quels logiciels avons-nous besoin?	14
L’installation typique	14
Installation sur Windows	16
Installation sur Mac OS	21
Installation sur Linux	22
 3 Structurer son premier document	 27

La compilation	28
Rédaction d'un document simple	31
Types de documents et caractères spéciaux	34
4 Les packages	39
La philosophie des packages	40
Comment s'en servir ?	40
Comment installer un package ?	43
 II Utilisation basique de L^AT_EX	 47
 5 Maîtriser sa mise en page (1/2)	 49
Structure des documents	50
Page de garde	54
Alignements de texte et sauts	57
 6 Maîtriser sa mise en page (2/2)	 61
La forme de votre publication	62
Marges et interlignes	63
Les listes	66
Les styles	70
 7 Les polices	 75
Graisse, style, taille	76
Couleur	79
Les packs de polices	81
 8 Les notes	 85
Citation, code brut et URL	86
Minipage et texte encadré	90
Notes de bas de page, références internes	92
 9 TP 1 – Mise en page de la Déclaration universelle des droits de l'homme	 97
Les consignes	98
La correction	105

Aller plus loin	117
---------------------------	-----

III Utilisation avancée de LaTeX 125

10 Les figures 127

Les différents formats d'images	128
Insertion d'images et de figures	129
Les flottants	137

11 Les tableaux 143

Structure type d'un tableau	144
Fusion de cellules	147
Autres paramètres applicables à un tableau	149

12 Les mathématiques 157

Saisir une expression mathématique	158
Fonctions usuelles	160
Les intégrales	162
Les systèmes d'équations	166
Les matrices	171
Mise en forme de théorèmes, lemmes...	175
Flèches, symboles étirables et espaces	177

13 Sommaire et index 181

Tables des matières	182
Tables des figures et tableaux	189
Les index	191

14 La bibliographie 197

Principe d'une bibliographie	198
La base de données	199
Mise en place de la bibliographie	202

15 TP 2 – Rédaction d'un article présentant le théorème de Pythagore 209

Les consignes	210
La correction	217

Être plus à l'aise avec L ^A T _E X	223
IV Annexes	229
A Les caractères spéciaux	231
Environnement mathématique	232
Autres caractères spéciaux	237
B Liste des packages	239
La liste	240
Petite astuce	242
C Les gabarits	243
La classe <code>article</code>	244
La classe <code>report</code>	245
La classe <code>book</code>	246
La classe <code>letter</code>	247
D Mémento	249
Les essentiels	250
La commande <code>\documentclass</code>	250
Les éléments de mise en page	251
Les polices	255
Les notes	258
Les figures	258
Les tableaux	259
Les mathématiques	260
Les tables	263
Index	263
Bibliographie	264
E Aller plus loin	265
Typographie et encodage	266
Les cours d'approfondissement	267
Les incontournables	268

Beamer	268
Les interfaces graphiques alternatives	268

Première partie

Découverte de L^AT_EX

Chapitre 1

Qu'est-ce que $\text{\LaTeX}$?

Difficulté : 

$\text{\LaTeX}$, vous en avez déjà entendu parler, mais ce nom vous intrigue. À quoi sert-il ? Est-il réellement utile ? Quels sont ses avantages et inconvénients ? Pourquoi a-t-il été créé, et quelle est son histoire ?

Toutes ces questions trouveront leurs réponses dans ce chapitre. Du rôle de $\text{\LaTeX}$ dans l'avancée du journalisme scientifique à ses caractéristiques techniques, en passant par un court portrait de ses créateurs, cette introduction vous apportera d'intéressantes informations culturelles et techniques. Bien entendu, nous verrons quelques exemples de la mise en page qu'il permet, afin de mieux comprendre les capacités de ce langage (car c'en est un). Et tout vous sera expliqué à partir de... zéro.



$\text{\LaTeX}$, quésaco ?



$\text{\LaTeX}$ se prononce « latec » ou « latèque », mais certainement pas « latex ». Il est important de vous en souvenir, sinon vous risquerez de provoquer de nombreux quiproquos avec vos collègues lorsque vous parlerez de $\text{\LaTeX}$.



FIGURE 1.1 – Le logo de $\text{\LaTeX}$

$\text{\LaTeX}$ est un langage créé pour séparer le fond de la forme lors de la création d'un document ou d'une publication. Plus clairement, l'auteur tape des instructions dans une sorte de bloc-notes¹ et structure son texte grâce à des mots et des commandes propres à $\text{\LaTeX}$. Par exemple, l'auteur peut indiquer à $\text{\LaTeX}$ de placer la première partie de son texte en gras, et une autre en italique. En somme, il **décrit** comment il veut hiérarchiser l'information. Ensuite, son code est traité par un logiciel : $\text{\LaTeX}$ choisit alors les meilleurs agencements et la disposition optimale pour chacun des éléments du document.

En résumé, $\text{\LaTeX}$ est un **langage de description** donnant à l'auteur les moyens d'obtenir des documents mis en page de façon professionnelle sans avoir à se soucier de leur forme. La priorité est donnée à l'essentiel : le contenu.



$\text{\LaTeX}$ est une arme à double tranchant, car s'il a tendance à générer une mise en page presque parfaite, il est souvent très difficile d'agencer les éléments de façon plus exotique².

Un soin tout particulier a été consacré à l'élaboration de commandes permettant d'écrire très facilement d'impeccables formules mathématiques. La figure 1.2 est un exemple de rendu d'une formule compliquée, qui n'a pourtant requis qu'une seule ligne de code $\text{\LaTeX}$. Cette spécificité et cette facilité d'écriture des formules font de $\text{\LaTeX}$ un outil très diffusé dans le monde scientifique.

1. Nous définirons précisément les outils et logiciels nécessaires à l'utilisation de $\text{\LaTeX}$ dans le prochain chapitre.

2. $\text{\LaTeX}$ est conçu pour créer des documents lisibles et beaux. Bien souvent, quand il essayera de changer votre mise en page (car pour lui, elle n'est pas adaptée), il aura raison.

$$\sum_{k=2}^{47} k + 1$$

FIGURE 1.2 – Une formule

Word versus L^AT_EX

Nous allons maintenant développer les différences entre L^AT_EX et Word à travers un comparatif global de leurs caractéristiques, suivi d'un résumé traitant des avantages que présente L^AT_EX.

Comparatif général

Que ce soit dans le cadre de votre travail ou de celui de vos études, vous avez sûrement déjà été confronté à la lourde et ingrate tâche que représentent l'écriture et la mise en page de longs rapports. Rédiger un document de plus de 15 pages demande déjà un grand travail et, bien souvent, la mise en page sur Word donne du fil à retordre.

La cause principale ? L'utilisateur doit gérer à la fois le fond et la forme sur un logiciel fonctionnant comme Word. Trop souvent, il s'enlise dans ces manipulations et perd du temps.

Word reste l'outil le plus développé à l'heure actuelle. La concurrence est rude entre les suites logicielles de bureautique, et il y a fort à parier qu'au fil des ans, L^AT_EX et Word se rapprocheront énormément, tant au niveau de la facilité d'utilisation que de l'optimisation de la prise en main.



L^AT_EX fait peur aux débutants à cause de son interface austère, mais Word n'aide pas ces derniers à réellement apprendre l'emploi correct de ses outils.

Ces deux problèmes et la manière dont ils seront résolus constitueront des facteurs prépondérants dans la progression respective des deux logiciels sur le marché de l'informatique.

Concrètement, les modifications deviennent vite un cauchemar une fois qu'on a sélectionné et modifié des dizaines de petits éléments. Cela n'est qu'un exemple, mais dans vos publications, vous avez sûrement déjà eu besoin de :

1. mettre en page des images ;
2. légender les figures ;

3. écrire des formules mathématiques (sous Word, c'est extrêmement fastidieux) ;
4. dessiner des schémas ;
5. créer des tableaux.

Autant de tâches indispensables, mais qui prennent du temps (même si elles sont faites dans les règles de l'art), à un point tel que nous pouvons avancer que, pour un utilisateur expérimenté, le temps consacré à la rédaction représente moins de 50 % du temps passé sur un long document.

C'est à partir de ce constat que naît l'intérêt de L^AT_EX. L^AT_EX n'est pas un traitement de texte du même type qu'OpenOffice et Word. OpenOffice et Word sont ce qu'on appelle des WYSIWYG (*what you see is what you get* = ce que vous voyez est ce que vous obtenez). Ils vous permettent de mettre en gras votre texte en le sélectionnant et en cliquant sur le bouton gras : vous voyez alors votre texte en gras.

La figure 1.3 vous montre la différence de présentation entre L^AT_EX (à gauche) et Word (à droite).

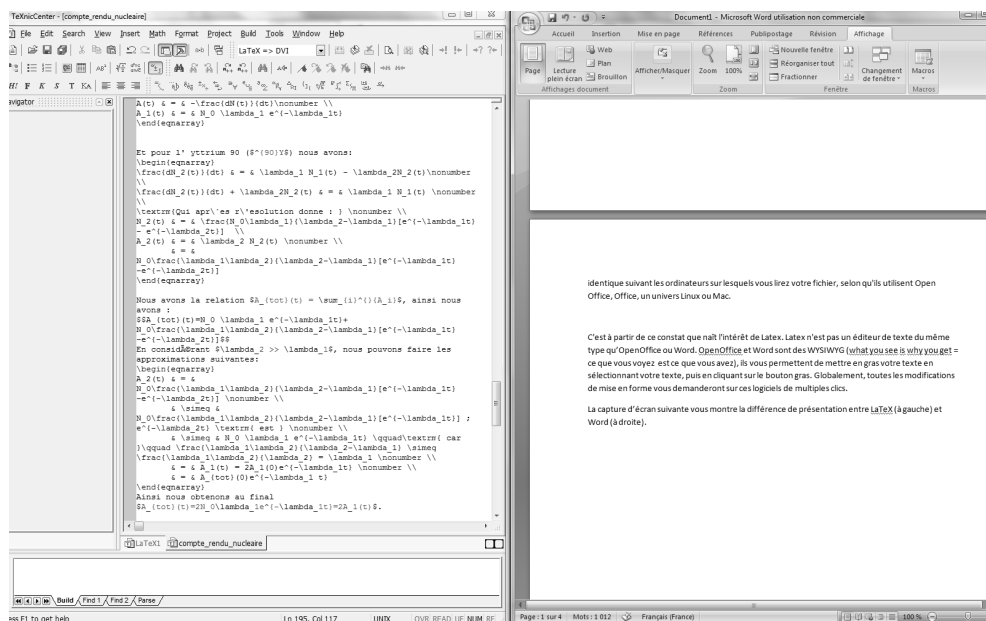


FIGURE 1.3 – L^AT_EX vs Word

J'insiste là-dessus : L^AT_EX ne se comporte et ne s'utilise pas de la même façon que Word. Une fois que vous aurez choisi quel type de publication vous écrivez (livre, article, C.V., lettre...), vous n'aurez plus qu'à taper votre texte au kilomètre dans son éditeur, en mettant en avant les paragraphes, les titres, ou les mots importants de la manière suivante (ici, un morceau de texte est mis en italique) :

La liberté consiste à pouvoir faire \textit{tout ce qui ne nuit pas} à autrui...

L^AT_EX, à partir de votre texte, créera une mise en page lisible et adaptée, mettra en valeur vos mots en gras, vos titres, vos images et légendes. De façon plus large, c'est un langage créé pour respecter les normes éditoriales et typographiques, qui vous permet d'écrire sans avoir à vous soucier de la mise en page. L'idée de taper des balises et des commandes en plein milieu du texte peut sembler étrange au début, mais vous découvrirez très vite qu'elle simplifie la vie.



L'optimisation de la création de documents typographiquement propres est l'argument principal jouant en faveur de l'utilisation de L^AT_EX. En deuxième rang, nous pourrions citer la gratuité, bien qu'OpenOffice soit également gratuit.

Une attention toute particulière a été apportée à la simplification de l'écriture des formules mathématiques et autres expressions scientifiques, souvent enfantées dans la douleur avec des outils comme l'éditeur d'équations de Word. Nous apprendrons bien entendu à maîtriser cette partie de L^AT_EX.



Information : il existe des éditeurs d'équations performants permettant de recopier des formules au sein de Word. Néanmoins, L^AT_EX intègre nativement ce genre d'outil et il n'est pas nécessaire d'installer de programme annexe pour effectuer des opérations sur un schéma ou une expression mathématique.

Les avantages de L^AT_EX

Il est possible que dans un élan de folie, un auteur estime un élément de mise en page peu adapté ou souhaite simplement changer la police d'un titre. Le voilà donc face à un problème déjà suggéré plus haut : comment modifier un style de titre présent sur 60, 100 voire 200 pages ? Si certains outils WYSIWYG ont eux aussi apporté une solution à ce problème, sachez qu'avec L^AT_EX, il vous sera possible de changer les caractéristiques de tous vos titres en une fois, de sorte que votre modification soit appliquée sur toutes les pages.

En réalité, avec L^AT_EX, tout est modifiable, tout est paramétrable. Le logiciel vous donnera une base par défaut propre et structurée pour mettre en page vos publications, mais vous serez libre de modifier les moindres marges, alignements et styles selon vos désirs. Que du bonheur !

Si nous devons lister d'autres qualités de L^AT_EX, nous pourrions citer en vrac :

- sa gestion des notes ;
- sa gestion des flottants (dont nous parlerons dans la troisième partie du livre) ;
- sa gestion des longs documents ;
- la possibilité qu'il offre de hiérarchiser facilement du texte en parties, chapitres, sections. . . ;
- la possibilité qu'il offre de faire des références (à un paragraphe, une figure. . .) de façon très simple et souple ;

- la possibilité qu'il offre de générer automatiquement des bibliographies, index et tables des matières.

$\text{\LaTeX}$ et la rétrocompatibilité

Rétrocompatibiliquoi ? La rétrocompatibilité, c'est la capacité qu'a un logiciel de produire des fichiers lisibles par ses versions précédentes qui n'ont pas été mises à jour. Vous avez sûrement déjà été confronté au problème causé par une différence de version avec une présentation sur PowerPoint. Si vous n'avez jamais eu le souci sous PowerPoint, vous l'avez probablement rencontré sous Word : un changement de PC peut miraculeusement décaler tous les paragraphes de votre cher rapport et rendre tous les schémas illisibles³. Ces deux problèmes sont dits de rétrocompatibilité, causés par des logiciels ne produisant pas des fichiers compatibles avec leurs versions précédentes.

Sur $\text{\LaTeX}$, ce genre de complication ne survient que très rarement⁴. Un document écrit sur un ordinateur A aura, dans la grande majorité des cas, exactement le même rendu sur un ordinateur B, quelle que soit la version de $\text{\LaTeX}$ utilisée et le système d'exploitation (Linux, Windows, Mac OS...). Ajoutons à cela qu'il est souvent possible de produire des documents en PDF avec $\text{\LaTeX}$ ⁵.

Suffisamment appréciable pour être noté, n'est-ce pas ?



Les suites logicielles classiques de traitement de texte intègrent depuis peu la fonction de création de PDF. L'initiative de $\text{\LaTeX}$ semble avoir eu du succès.

$\text{\LaTeX}$ et les publications

Nous venons de traiter les points qui font de $\text{\LaTeX}$ un langage de qualité, à savoir :

1. la rétrocompatibilité ;
2. la création de PDF ;
3. la possibilité de lire ses documents sur tous les systèmes d'exploitation ;
4. la présence d'outils permettant la modification rapide de l'ensemble des éléments formant la mise en page ;
5. la possibilité de passer (enfin) plus de temps à écrire du contenu qu'à le mettre en forme.

Autant d'avantages qui ont fait de $\text{\LaTeX}$ le chouchou du monde de l'édition et du monde scientifique... Tant et si bien que bon nombre d'ouvrages sont édités en $\text{\LaTeX}$, des

3. L'expérience prouve que l'individu lambda se sent très bête dans ce genre de cas.

4. Il peut arriver qu'une extension du logiciel, que nous appellerons **package**, entre en conflit avec une de ses versions précédentes. Néanmoins, cela reste exceptionnel.

5. Le PDF est un format international de mise en page, connu pour ne pas déformer les documents et présenter le même rendu à peu près sur toutes les machines.

professionnels de ce langage se faisant payer par des maisons d'édition pour mettre en page des livres de manière irréprochable. Nous allons nous pencher sur les applications de ce langage dans les publications et le milieu scientifique.

L^AT_EX dans les publications scientifiques



FIGURE 1.4 – Les Livres du Zéro sont écrits en L^AT_EX

Point important : les scientifiques communiquent essentiellement entre eux grâce à des articles. Ces articles sont la plupart du temps écrits en L^AT_EX. Ces publications sont légion et chaque semaine, des dizaines de revues scientifiques trient, sélectionnent et mettent en page des articles traitant des innovations et découvertes récentes. En somme, si la science fait avancer le monde, L^AT_EX est à l'heure actuelle son format de prédilection. D'ailleurs sachez que les « Livres du Zéro » sont écrits en L^AT_EX⁶ (figure 1.4), comme la majorité des fascicules et livres de cours que vous avez pu consulter par le passé.

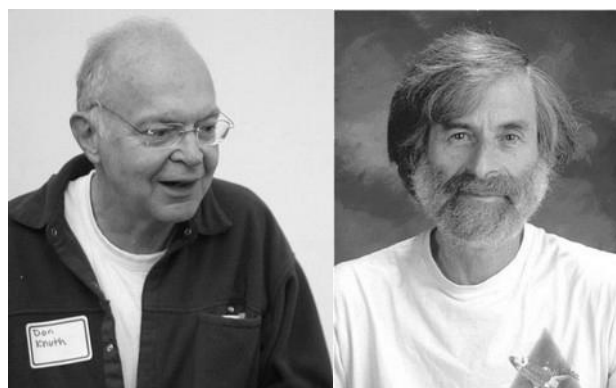
L^AT_EX dans le monde de l'éducation

Je suis actuellement étudiant à l'université de Bordeaux 1 : sachez qu'à partir de la première année de Master, dans certaines matières (notamment la physique et les maths), il est obligatoire de rendre ses rapports de stage, mémoires et autres textes au format L^AT_EX. Il est également utilisé de façon extrêmement fréquente dans les laboratoires de recherche grâce à l'aspect pratique du format PDF (il est facile à exporter et à lire sur les ordinateurs de tous les pays).

L^AT_EX et les présentations

Notez que L^AT_EX permet non seulement de faire des rapports, mais aussi de créer des présentations contenant des *slides* (diapositives) de qualité incomparable, et bien plus encore ! Mais nous en parlerons plus loin dans ce livre.

6. Y compris ce livre que vous tenez dans les mains !



(a) Donald Knuth

(b) Leslie Lamport

FIGURE 1.5 – Les créateurs

Historique rapide

Difficile d'enseigner un langage informatique sans survoler rapidement les raisons de sa création et les C.V. (généralement fort bien remplis) de ses pères. Nous allons ici développer l'historique de $\text{\LaTeX}$.

Tout commence avec la création du langage $\text{\TeX}$ par Donald Erwin KNUTH (chercheur américain en informatique, né en 1938), déclenchée en 1977 par un ras-le-bol de l'inélégante mise en page de ses articles mathématiques publiés par l'*American Mathematical Society*.

Comme souvent dans le monde informatique, un besoin a été la raison de l'innovation, en l'occurrence : $\text{\TeX}$. Le langage avait été conçu non pas pour produire de beaux documents, mais pour accroître la lisibilité et optimiser l'insertion de formules mathématiques.

Nous ne sommes pas des typographes, ni des experts de la mise en page, mais ceux qui ont participé à la création de $\text{\TeX}$ le sont, et ils ont créé un ensemble cohérent de commandes permettant à l'utilisateur d'obtenir un résultat professionnel.

$\text{\LaTeX}$ est une surcouche de $\text{\TeX}$ et n'a été créé que plus tard, plus exactement en 1985 par Leslie LAMPORT (chercheur en informatique américain, né en 1941). $\text{\TeX}$, bien que révolutionnaire, était relativement compliqué à utiliser. $\text{\LaTeX}$ permit une simplification énorme de bon nombre de tâches grâce à des **macros** intégrées au programme.



Une macro est une commande qui vous permet de lancer une série d'actions à votre place. Cela peut aller du double-clic à des processus bien plus compliqués. Ici, nous n'allons pas nous en préoccuper : les macros sont intégrées à $\text{\LaTeX}$ pour nous simplifier la vie, ne nous la compliquons pas.

La dernière évolution majeure est $\text{\LaTeX}_{2\epsilon}$, c'est celle avec laquelle nous allons travailler

dans ce cours.

L^AT_EX est un logiciel libre, vous pouvez donc à votre guise le distribuer, vous amuser à modifier son code (pour autant que cela vous amuse), vendre des millions de copies dans l'optique de dominer le monde. . . du moment que votre logiciel ne reprend pas le nom « T_EX » (c'est écrit dans la licence de T_EX).



Ce côté « bidouillable » de L^AT_EX a entraîné l'apparition de nombreux logiciels permettant la création de documents, abondance qui laisse parfois le néophyte perplexe. J'en ai choisi quelques-uns pour vous et vous guiderai dans leur installation, que vous soyez sous Mac OS, Windows, ou Linux ! Tout cela se passe dans le prochain chapitre.

En résumé

- L^AT_EX est un langage de description libre et gratuit. Il permet de concevoir des documents de qualité professionnelle sans connaissances en typographie et mise en page.
- Contrairement à un traitement de texte comme Word, L^AT_EX vous permet de vous concentrer sur le contenu de votre document. Tout le reste est généré automatiquement par L^AT_EX. Pas besoin de se préoccuper de la numérotation des pages, de la création d'un sommaire, de la numérotation des figures ou encore des marges et alinéas !
- L^AT_EX permet de produire des documents PDF qui s'affichent de la même façon sur tous les ordinateurs, qu'ils soient sous Windows, Mac OS ou Linux.
- C'est un langage très populaire dans les études supérieures, chez les scientifiques et dans le monde de l'édition. Il excelle en particulier dans l'écriture de formules mathématiques, domaine dans lequel il fait figure d'outil de référence.
- Ce livre a été écrit en L^AT_EX. ;-))

Chapitre 2

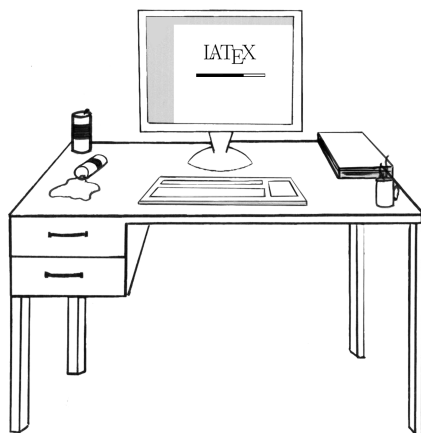
Installer L^AT_EX

Difficulté : 

Le chapitre précédent vous a permis de vous familiariser avec les caractéristiques de L^AT_EX et quelques-unes des raisons qui ont fait son succès.

Ce deuxième chapitre traite de façon détaillée de l'installation des logiciels permettant d'utiliser L^AT_EX sur votre machine. L^AT_EX étant libre, de nombreux environnements graphiques ont été développés, le choix est vaste et vous pourrez trouver des dizaines de méthodes et de logiciels permettant d'installer un éditeur L^AT_EX.

N'ayez crainte, les environnements sont différents, mais à l'intérieur, ces logiciels utilisent le même noyau dur¹. En conséquence, un texte tapé sous Windows via un éditeur quelconque pourra être lu et modifié avec d'autres logiciels et environnements (par exemple sur des logiciels conçus pour Mac ou Linux).



1. T_EX, dont nous avons parlé dans le chapitre 1.

De quels logiciels avons-nous besoin ?

Dans cette sous-partie, notre regard se porte sur ce que nous souhaitons obtenir grâce à L^AT_EX. En l'occurrence, nous souhaitons (et parviendrons à) créer des documents en PDF et en PostScript.



Qu'est-ce que PostScript ?

C'est un format de fichier rendant possible sa lecture par des imprimantes sans l'intermédiaire d'un PC, chose que les PDF ou les documents générés par Word ne permettent pas. Offrant beaucoup d'avantages, il est très utilisé au sein des laboratoires de recherche. Vous n'aurez normalement pas grand besoin d'utiliser des fichiers au format PostScript pour vos publications, néanmoins, certains chercheurs et éditeurs en sont de grands fans et ne jurent que par lui. Nous apprendrons donc à manipuler ce format.

L'installation typique

Une installation classique de L^AT_EX comporte trois éléments.

- **Une distribution L^AT_EX** : c'est un logiciel comportant toutes les composantes de L^AT_EX, c'est lui qui va transformer via L^AT_EX votre code tapé dans l'éditeur L^AT_EX en un document PDF ou PostScript. Vous l'aurez compris, c'est le morceau le plus important.
- **Un ou plusieurs lecteurs de PostScript et de PDF**, pour pouvoir lire et imprimer vos productions.
- **Un éditeur L^AT_EX** : c'est un logiciel vous permettant de taper votre texte, de mettre en couleur les différents codes utilisés, de numérotter vos lignes, etc. En somme, choisir un éditeur L^AT_EX vous simplifie la vie, mais **il n'est pas indispensable** : il est possible de taper du L^AT_EX dans un simple éditeur de texte (même si cela est moins pratique). Il propose aussi bon nombres de raccourcis et boutons utiles à la création de document ².

Comme dit ci-dessus, vous pouvez choisir de ne pas utiliser d'éditeur L^AT_EX et d'employer à la place un éditeur de type Bloc-notes, ou de faire usage d'outils de coloration de texte plus évolués. Les éditeurs L^AT_EX colorent le code (ce qui le rend beaucoup plus lisible) et vous permettent, via des raccourcis et des fenêtres vous informant des bugs, de faciliter la production de documents. De la figure 2.1 à la figure 2.3, vous pouvez voir le même code, tapé respectivement dans un éditeur de texte simple (Bloc-notes), dans un éditeur de texte avec coloration (Notepad++ ³) et pour finir, dans un éditeur L^AT_EX (T_EXnicCenter).

Par la suite, nous utiliserons un éditeur L^AT_EX.

2. Ces boutons et raccourcis diffèrent d'un éditeur à l'autre.

3. Les habitués de la programmation web auront sûrement beaucoup d'affinités avec ce logiciel.



FIGURE 2.1 – Bloc-notes



FIGURE 2.2 – Notepad ++

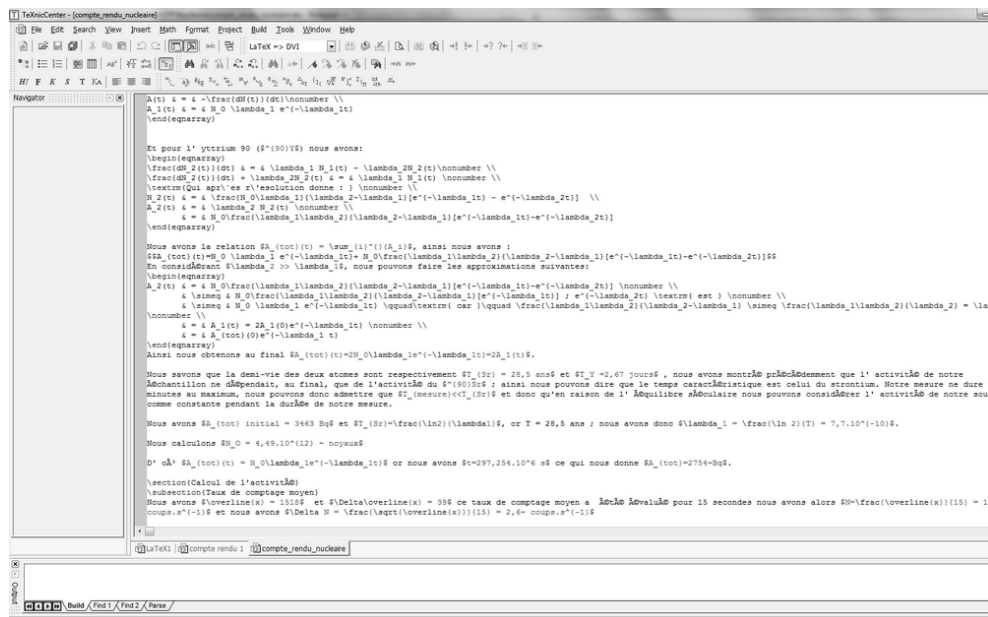


FIGURE 2.3 – T_EXnicCenter

Installation sur Windows

Notre installation logicielle pour L^AT_EX sur Windows se déroulera en trois temps, je vais vous les détailler.

MiK_TE_X, la distribution L^AT_EX

Tout d’abord nous allons installer **MiK_TE_X**, une des distributions L^AT_EX disponibles pour Windows. Pour l’installer, la procédure est plutôt simple, mais elle vous demandera de la **minutie**.

- Créez un dossier appelé **Latex** sur votre bureau ou dans votre dossier **Mes documents**, il servira à ranger la masse de fichiers que MiK_TE_X utilise pour son fonctionnement interne.
- Allez sur le site de MiK_TE_X ⁴.

▷ Site web de MiK_TE_X
Code web : 106176

Dans la colonne de gauche, vous trouverez un titre « **Download** » (figure 2.4) avec en dessous un lien vers la page de téléchargement de la dernière version de MiK_TE_X, **cliquez dessus**.

- Dans la nouvelle page, vers le bas, vous trouverez un paragraphe intitulé « Installing the complete MiK_TE_X system » comportant un bouton **download**, vous devez cli-

4. Pour savoir comment utiliser les codes web, reportez-vous à l’avant-propos au début de ce livre.

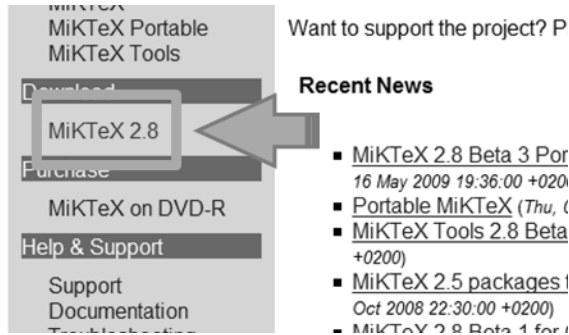


FIGURE 2.4 – Lien de téléchargement de MiKTeX

quer dessus et enregistrer le fichier dans votre dossier **Latex**. Télécharger ce fichier devrait être rapide, il pèse environ 3 Mo.



Le fichier que vous avez téléchargé est ce qu'on appelle un gestionnaire d'installation. Dans notre cas, il va dans un premier temps télécharger MiKTeX, et dans un second temps il l'installera.

- **Lancez le gestionnaire d'installation** (normalement vous l'avez enregistré dans votre dossier **Latex**, si vous l'avez enregistré ailleurs, pas d'inquiétude, ça ne change rien à la procédure) en **double cliquant** dessus.
- **Cochez** la case « I accept the MiKTeX copying conditions » puis cliquez sur **Suivant** (figure 2.5).

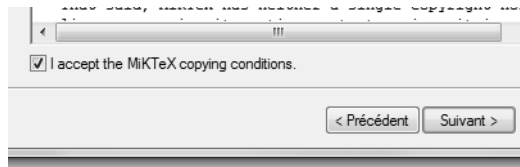


FIGURE 2.5 – Case à cocher

- **Cochez** « Download MiKTeX » puis cliquez sur **Suivant**.
- **Cochez** « Complete MiKTeX » puis cliquez sur **Suivant**.
- **Choisissez** un serveur de téléchargement dans un pays plutôt proche du vôtre (par exemple, j'ai sélectionné un serveur français au hasard). Plus le serveur de téléchargement choisi est proche, plus vous chargerez rapidement. Cliquez sur **Suivant**.
- **ÉTAPE IMPORTANTE !** MiKTeX vous demande alors où vous voulez qu'il enregistre tout son attirail. C'est ici que le dossier **Latex** préalablement créé montre tout son intérêt. **Cliquez sur Browse** (Parcourir, en français), cherchez votre dossier **Latex** puis cliquez sur **OK**. Normalement, votre écran doit ressembler à la capture d'écran montrée en figure 2.6.
- Cliquez sur **Suivant**, vous aurez droit à un petit écran récapitulatif comprenant le

- nom du serveur que vous avez choisi pour télécharger et le dossier de destination des fichiers de MiKTeX (ici le dossier **Latex**).
- Appuyez sur **Start** pour lancer le téléchargement.

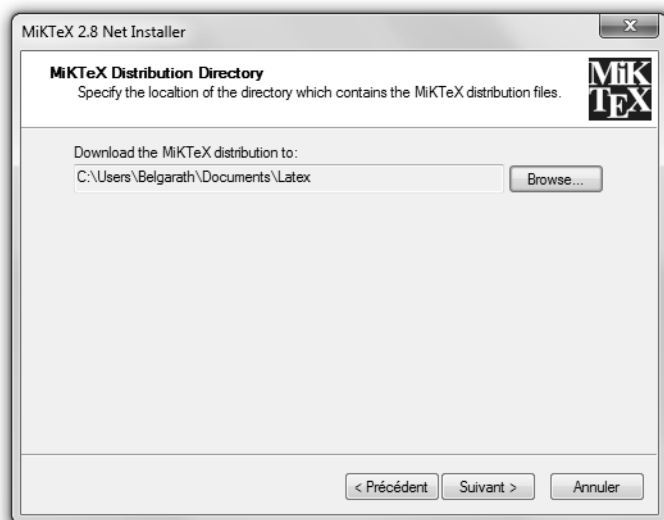


FIGURE 2.6 – Choix du dossier de destination



MiKTeX doit télécharger plusieurs centaines de Mo de fichiers, et cela peut prendre pas mal de temps. Profitez donc de ces instants de répit pour tuer des zombies dans un jeu quelconque ou refaire le monde autour d'un café. La folle aventure de l'installation ne reprendra que dans quelques minutes. Soyez rassuré, le plus dur est derrière vous.



Il est possible qu'à un moment ou à un autre, l'installation de MiKTeX plante. Il vous faudra alors tout recommencer depuis le début (ça m'est arrivé lors de la rédaction de ce livre). Ce n'est pas la grande joie d'appuyer cinquante fois sur **Suivant**, mais une bonne installation vous permettra de travailler sereinement plus tard.

- La longue période de chargement est terminée, l'assistant d'installation vous félicite pour votre patience. **Fermez** la fenêtre et **relancez le gestionnaire d'installation**. Une nouvelle fois, vous allez **cocher** la case « I accept the MiKTeX copying conditions » puis cliquer sur **Suivant**. Ensuite, vous allez choisir « **Install MiKTeX** »

- (enfin) et cliquer sur **Suivant**.
- Sur l'écran qui s'affiche alors, le gestionnaire d'installation vous demande si vous souhaitez installer la version basique ou complète. Vu que nous avons téléchargé la version complète, nous cocherons donc « **Complete MiKTeX** » avant de cliquer sur **Suivant**.
 - L'écran suivant vous demande si vous souhaitez installer MiKTeX pour vous seul ou sur tous les comptes présents sur votre machine. Faites selon vos souhaits puis cliquez sur **Suivant**.
 - **ÉTAPE IMPORTANTE** : MiKTeX vous demande où sont rangés tous les fichiers que vous avez téléchargés plus tôt. **Vérifiez** bien que le bon dossier est sélectionné avant de cliquer sur **Suivant**. L'écran qui suit vous montre où MiKTeX va s'installer (les fichiers que vous avez téléchargés dans la première partie sont ceux dont il se sert pour l'installation). Vous pouvez changer la destination si vous voulez avant de cliquer sur **Suivant** et d'atterrir sur un écran vous demandant le format de papier par défaut que vous utiliserez (ici A4, vu que nos imprimantes impriment surtout en A4⁵). Cet écran vous proposera aussi d'activer la demande de permission à chaque fois que le programme devra charger un package manquant. Laissez le réglage par défaut et appuyez sur **Suivant**.
 - Un écran vous résume vos choix, et une fois que vous aurez cliqué sur **Start**, la **procédure s'achèvera et le logiciel s'installera tranquillement**. N'oubliez pas de cliquer une dernière fois sur **Suivant** pour voir encore un merveilleux écran de félicitations. :-D



Cette étape est à la fois la plus cruciale et la plus ennuyeuse de l'apprentissage de $\text{\LaTeX}$. Lorsque j'ai dû l'installer la première fois, me repérer entre toutes les distributions et les outils n'a pas été évident. C'est pour cela que je vous ai détaillé au maximum la procédure d'installation, pour que vous ne viviez pas le même cauchemar. Si vous voyez d'autres utilisateurs de $\text{\LaTeX}$ sur Windows utiliser d'autres logiciels, pas de panique, rappelez-vous que $\text{\LaTeX}$ étant libre, il existe des multitudes de logiciels pour l'utiliser.

Ghostscript et Adobe Reader, vos lecteurs de PostScript et de PDF

Il est temps d'installer un lecteur de PostScript ainsi qu'un lecteur de PDF sur votre ordinateur, de sorte que vous puissiez :

- manipuler les fichiers PS (diminutif de PostScript) ;
- transformer les fichiers `.ps` en `.pdf` ;
- lire les `.ps` ;
- lire les `.pdf`.

5. Vous pouvez le modifier si vous avez des besoins très spécifiques.

Ghostscript

Pour ce qui est des PS, téléchargez le logiciel **Ghostscript** sur telecharger.com grâce au code web ci-dessous.

▷

Télécharger Ghostscript
Code web : 806251

Lancez le gestionnaire d'installation, cliquez sur **Setup**, vérifiez que l'emplacement où il propose de s'installer vous plaît puis cliquez sur **Install**.

Adobe Reader

Les PDF se lisent la plupart du temps avec **Adobe Reader** (mais il existe beaucoup d'autres logiciels capables de lire des PDF). Ce logiciel est souvent présent par défaut sur les ordinateurs de bureau. Si ce n'est pas le cas, vous pourrez l'obtenir lui aussi sur telecharger.com.

▷

Télécharger Adobe Reader
Code web : 239113

Sa procédure d'installation est aussi simple que celle de **Ghostscript**, elle ne comporte aucun piège.

Bravo ! Vous venez d'installer Ghostscript et Adobe Reader.

T_EXnicCenter, l'éditeur L^AT_EX

T_EXnicCenter sera votre éditeur L^AT_EX, vous avez d'ailleurs pu voir une capture d'écran de ce programme dans le paragraphe traitant des logiciels nécessaires. Pour l'installer, il vous faudra d'abord aller le chercher sur le site officiel.

▷

Télécharger TeXnicCenter
Code web : 759295

C'est le premier lien, nommé « **T_EXnicCenter Installer** » qui nous intéresse, **cliquez** dessus, il vous permettra de lancer le téléchargement du gestionnaire d'installation. Une fois que c'est fait, démarrez-le.

- À l'écran d'accueil, appuyez sur **Next**.
- Cochez « **I accept the agreement** » puis cliquez sur **Next**.
- **T_EXnicCenter** vous propose de s'installer dans un dossier, changez-le s'il ne vous convient pas puis cliquez sur **Next**.
- Laissez l'installation sur **Typical** et cliquez sur **Next**.
- Il vous propose de créer un raccourci dans le menu « Démarrer », modifiez selon vos souhaits puis cliquez sur **Next**.
- Il vous propose de créer une icône sur le bureau, modifiez si besoin est avant de cliquer sur **Next**.
- S'affiche alors le résumé, vous pouvez cliquer sur **Install** après avoir vérifié les informations qu'il contient. Le petit écran de félicitations apparaît et avec lui, la possibilité

de cliquer sur le bouton **Finish**. ;-)

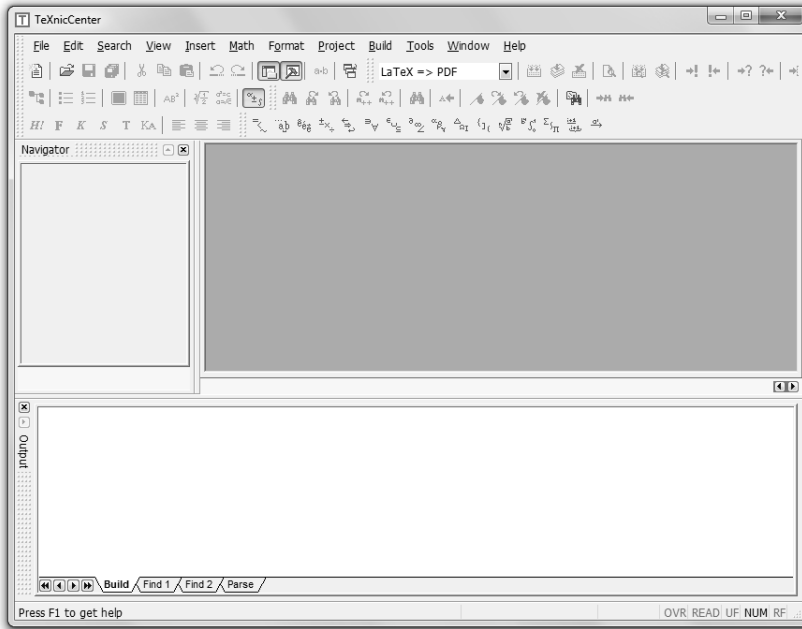


FIGURE 2.7 – T_EXnicCenter



L'installation est achevée, vous faites désormais cohabiter sur votre PC les trois composantes nécessaires à votre apprentissage de L^AT_EX. Installation fastidieuse, mais importante pour la suite! Nous ferons nos premiers tests dans le chapitre suivant et configurerons T_EXnicCenter.

Installation sur Mac OS

Utilisateurs de Mac, vous êtes de grands chanceux.

Votre système d'exploitation remporte la palme de l'installation L^AT_EX la plus simple et agréable. Les détenteurs de Windows viennent de souffrir environ trois heures pour installer toutes les composantes de L^AT_EX séparées en trois alors que pour vous, tout est rassemblé dans un seul fichier!

Le code web fourni ci-dessous mène au site du projet **MacT_EX**, au sein duquel se trouve un énorme fichier comportant tous les outils utiles à la réalisation d'un projet L^AT_EX. Le fichier à télécharger se trouve en plein milieu de la page (figure 2.8), téléchargez-le sur votre Mac et lancez le gestionnaire d'installation avant de suivre la procédure d'une simplicité enfantine qui vous est proposée.

▷ Télécharger MacTeX
Code web : 846763

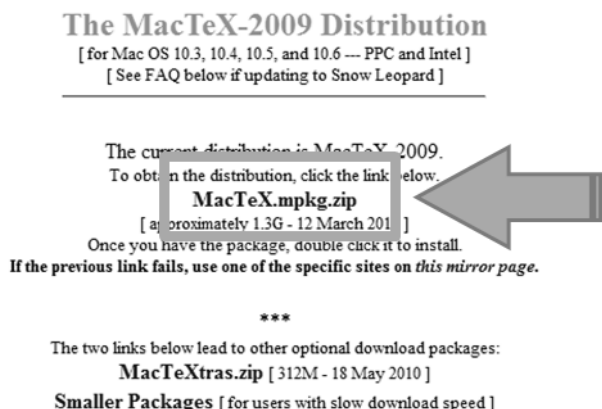


FIGURE 2.8 – Installation de MacTeX

Une fois cela fait, vous pourrez démarrer $\text{\TeX}$ Shop, l'éditeur $\text{\LaTeX}$ pour Mac que nous utiliserons par la suite. Il devrait s'être installé dans votre dossier **Applications**, libre à vous de faire des raccourcis sur votre bureau.

Une capture d'écran de ce à quoi pourra ressembler votre $\text{\TeX}$ Shop vous est montrée à la page suivante (figure 2.9).



Avec MacTeX, vous avez installé une suite logicielle complète centrée autour de $\text{\LaTeX}$. Elle vous permettra par exemple d'écrire des équations en $\text{\LaTeX}$ que vous pourrez inclure par copier-coller dans des documents Word. Outils pratiques, ils ne seront pas abordés ici mais ils méritent que vous les testiez lorsque vous vous sentirez à l'aise avec $\text{\LaTeX}$.

Installation sur Linux

Dans le cadre de cet ouvrage, l'éditeur $\text{\LaTeX}$ choisi sera **Kile** (figure 2.11) et la distribution $\text{\LaTeX}$ se nomme **$\text{\TeX}$ Live**. Les méthodes d'installation sont pour ce système-ci extrêmement variées et les retours sur la facilité d'installation sur Linux ont été très mitigés. La majeure partie des étudiants utilisateurs de $\text{\LaTeX}$ sous Linux ont témoigné avoir trouvé beaucoup plus évidente et adaptée l'installation via le gestionnaire de paquets (grâce auquel vous téléchargerez dans un premier temps **$\text{\TeX}$ Live** puis **Kile**). Nous installerons dans le même temps de quoi lire vos fichiers PDF ou PostScript produits.

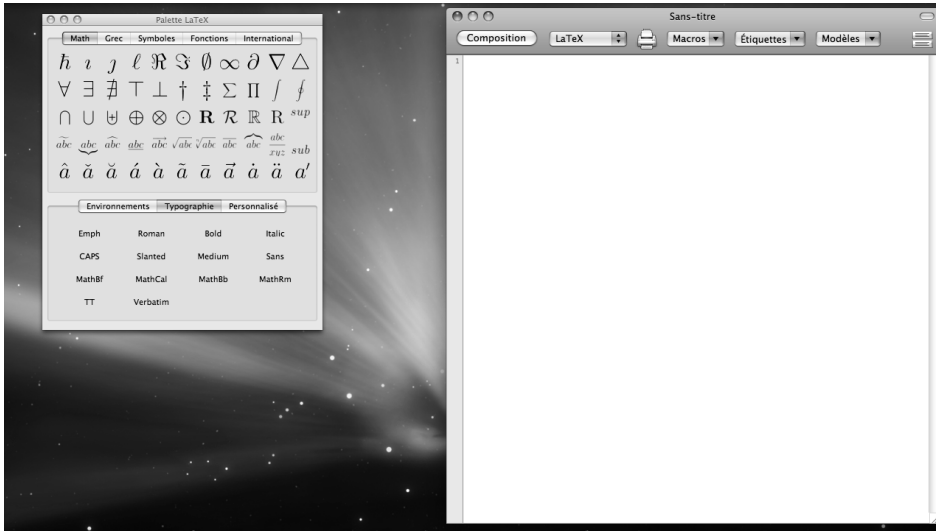


FIGURE 2.9 – TExShop



Beaucoup, à leurs débuts sous Linux, ont eu du mal avec l'installation de $\text{\LaTeX}$. N'hésitez pas à chercher dans vos groupes de travail ou votre entourage des personnes prêtes à vous aider. Il sera parfois plus évident pour vous de travailler dans les mêmes environnements que vos collaborateurs, et ce que vous lisez ici s'applique à n'importe quel éditeur $\text{\LaTeX}$, seul l'emplacement des boutons changera si vous utilisez un éditeur $\text{\LaTeX}$ différent.

Comme mentionné plus haut, nous allons utiliser le gestionnaire de paquets (figure 2.10). Dans un premier temps, **ouvrez-le**. Recherchez et installez **tous les paquets** contenant les mots suivants :

- **texlive** ;
- **ec-fonts-mftraced** (uniquement si vous êtes sous Mandriva) ;
- **cm-super** (pack de polices) ;
- **xdvi** (pas pour Ubuntu) ;
- **gv** (pour lire et manipuler les **.ps**) ;
- **spdf** ;
- **kile** (toujours en dernier).

Une fois ces multiples chargements effectués, vous pouvez vous reposer : votre installation fonctionne. Voici une capture d'écran du gestionnaire de paquets et une seconde de **Kile**, l'éditeur $\text{\LaTeX}$ avec lequel nous allons nous exercer et apprendre à utiliser $\text{\LaTeX}$.



Kile est un éditeur $\text{\LaTeX}$ facilitant l'écriture et la compilation de fichiers $\text{\LaTeX}$. Il est néanmoins possible de manipuler via les lignes de commande des fichiers $\text{\LaTeX}$ tapés dans un éditeur de texte. Nous apprendrons à nous servir de celles-ci dans le prochain chapitre (l'apprentissage des deux méthodes vous permettra par la suite d'utiliser selon vos souhaits la console ou l'interface graphique).

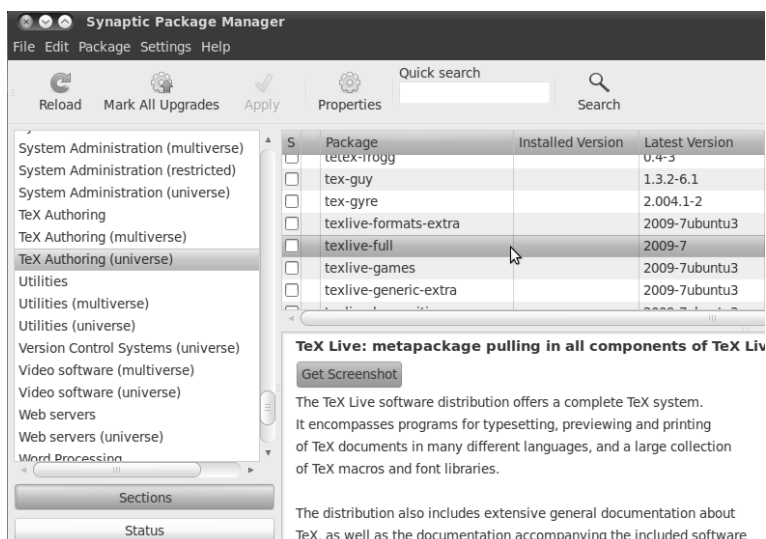


FIGURE 2.10 – Gestionnaire de paquets



FIGURE 2.11 – Kile



En cas de problème, veuillez consulter les sites internet de $\text{\TeX}$ Live ainsi que de Kile afin d'obtenir les informations spécifiques à votre installation Linux. Ce sont des outils très utiles qui vous permettront d'adapter la procédure d'installation à votre distribution Linux si jamais elle est un tant soit peu exotique.

En résumé

- Il existe trois grandes familles de logiciels utilisés pour concevoir des documents en $\text{\LaTeX}$: les **distributions**, les **éditeurs $\text{\LaTeX}$** et les **lecteurs**.
- Un éditeur $\text{\LaTeX}$ n'est pas indispensable à l'apprentissage de $\text{\LaTeX}$ ⁶.
- Le format de prédilection que nous choisirons par la suite sera le **PDF** : il est lisible sur quasiment tous les ordinateurs récents. Nous nous souviendrons néanmoins de l'attrait que suscite le format de fichier **PostScript** pour certains laboratoires de recherche.
- **Kile**, **$\text{\TeX}$ Shop** et **$\text{\TeX}$ nicCenter** sont actuellement très utilisés dans le monde de la recherche. Nous utiliserons ces éditeurs $\text{\LaTeX}$ sur des ordinateurs équipés respectivement de Linux, Mac OS et Windows.

6. Mais en utiliser un simplifie énormément l'apprentissage grâce à la coloration du code et aux différents outils disponibles.

Chapitre 3

Structurer son premier document

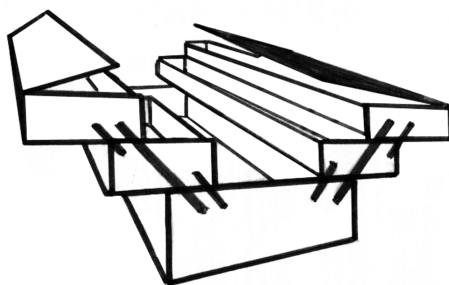
Difficulté : 

Durant les deux premiers chapitres de ce cours, vous avez découvert l'histoire de $\text{\LaTeX}$ ainsi que sa nature, et comment effectuer son installation. Ce nouveau chapitre sera pour vous l'occasion d'un premier contact avec les logiciels que vous venez d'installer.

Soyez sûr d'une chose : le plus dur est derrière vous.

Ce chapitre vous permettra d'écrire votre premier document, de l'imprimer et de comprendre comment $\text{\LaTeX}$ transforme un code informatique ingrat en un document chatoyant.

C'est un chapitre extrêmement important, car il explique les mécanismes essentiels de $\text{\LaTeX}$.



La compilation

Cette partie développe le concept de compilation et vous permettra d’appréhender le fonctionnement global de $\text{\LaTeX}$.

Cette portion du cours s’appuie sur un constat : des lignes de code se transforment en une mise en page. Vous pouvez voir sur la capture d’écran ci-dessous (figure 3.1) un document écrit via un éditeur $\text{\LaTeX}$ ainsi que son résultat une fois passé à la moulinette par votre distribution $\text{\LaTeX}$.

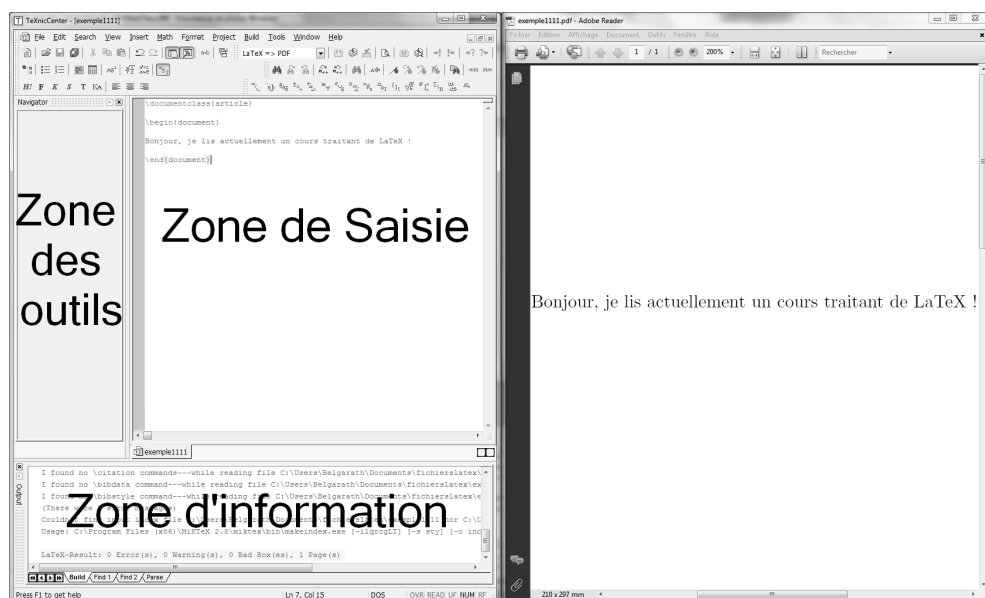


FIGURE 3.1 – Éditeur $\text{\LaTeX}$ et lecteur PDF



Que s’est-il passé ?

Souvenons-nous du chapitre précédent : nous y avons appris que les éditeurs $\text{\LaTeX}$ n’étaient pas indispensables, ils ne jouent qu’un rôle pratique. En revanche, nous avons fortement insisté sur le caractère incontournable de l’installation d’une distribution $\text{\LaTeX}$. La distribution est la pièce centrale, elle permet de transformer un fichier de lignes de code (ces fichiers sont des $\text{\LaTeX}$ en un fichier imprimable de type PS ou PDF.

Cette transformation s’appelle une **compilation**. Votre distribution $\text{\LaTeX}$ transforme votre fichier $\text{\LaTeX}$ en un fichier que vous pourrez par la suite utiliser pour vos travaux et publications.

Résumons tout cela.

1. Le code est écrit dans l'éditeur $\text{\LaTeX}$.
2. La distribution $\text{\LaTeX}$ le compile.
3. Le fichier `.tex` de départ se voit transformé en un fichier `.ps`, `.pdf` ou `.dvi` (un format très peu utilisé).

Le processus de création n'est pas plus compliqué que cela. Pour créer un document, il vous suffira d'écrire votre code au sein de l'éditeur $\text{\LaTeX}$ et votre distribution $\text{\LaTeX}$ se chargera du travail de transformation. Pour que le travail d'écriture du code et de compilation soit plus simple, les éditeurs $\text{\LaTeX}$ sont dotées d'une zone réservée aux raccourcis (pour les caractères spéciaux, par exemple), qui vous permettront d'insérer facilement dans vos écrits des morceaux de code dont vous vous servirez souvent ¹. Une seconde zone, dédiée aux informations liées à la compilation, vous permet de suivre les compilations et d'obtenir des informations sur d'éventuels bugs présents dans vos compositions. En somme, $\text{\LaTeX}$ communique avec vous pour vous aider à travailler.



Manipulerons-nous uniquement des fichiers `.tex`, `.dvi`, `.ps` et `.pdf` avec $\text{\LaTeX}$?

Pour rédiger un livre, il faut une bibliographie, un index et tout un tas d'éléments que $\text{\LaTeX}$ stocke dans des fichiers aux extensions différentes. Pendant la compilation, votre fichier `.tex` de départ créera donc une multitude de fichiers correspondant à ces besoins. Ne prenez pas peur si votre dossier ne contenant au départ qu'un `.tex` se voit tout à coup envahi par de multiples fichiers aux extensions étranges.

Nous allons résumer toutes ces informations via un schéma (figure 3.2). Je vous conseille vivement de bien le lire, il est souvent pratique de l'avoir à l'esprit.

Si vous avez compris ce schéma, sachez que $\text{\LaTeX}$ n'a pas un fonctionnement plus compliqué que cela. Il ne nous reste plus qu'à prendre en main les logiciels et à apprendre comment mettre en forme nos textes en utilisant les commandes et le formalisme de $\text{\LaTeX}$.

1. Nous verrons par la suite que certains accents doivent être traduits en code, et recopiés de façon très fréquente.

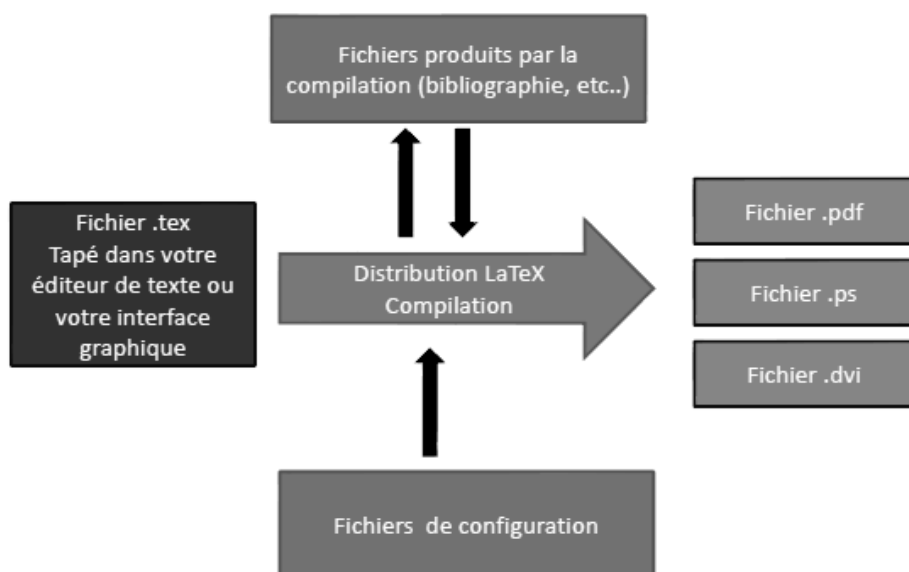


FIGURE 3.2 – Le processus de compilation

Rédaction d'un document simple

Dans cette partie, je vais vous aider et vous guider dans l'écriture de votre premier document. Il sera très simple et ne comportera qu'une phrase : « Bonjour, je lis actuellement un cours traitant de L^AT_EX ! »

Rédaction du fichier .tex

Dans un premier temps, il vous faut ouvrir votre **éditeur L^AT_EX** (Kile, T_EXShop, T_EXnicCenter ou autre). Il est possible que celui-ci vous pose quelques questions simples (pour connaître l'emplacement de votre lecteur de PDF, par exemple), répondez-y.



Si T_EXnicCenter vous demande le chemin d'installation de MiK_TE_X, vous devrez fournir un chemin du type
C:\Program Files (x86)\MiK_TE_X 2.8\miktex\bin.

Votre éditeur est à présent ouvert et vous vous trouvez face à une fenêtre scindée en trois parties : une partie console dans laquelle apparaissent les messages d'information (ainsi que les messages d'erreur), une zone réservée aux barres d'outils, et un espace réservé à la rédaction de votre fichier.

Grâce aux raccourcis disponibles dans la barre de menus, créez un nouveau document. Cliquez à l'intérieur de la zone réservée à l'édition et recopiez-y le paragraphe ci-dessous.

```
\documentclass{article}

\begin{document}
Bonjour, je lis actuellement un cours traitant de LaTeX !
\end{document}
```

Enregistrez ce fichier dans le dossier vide de votre choix en utilisant la barre de raccourcis présente en haut de votre éditeur. Nous l'appellerons `bonjour.tex`.

Il est interdit de mettre des espaces ou des accents dans le nom de vos fichiers. Votre compilateur n'apprécierait pas l'originalité et aurait tendance à vous renvoyer des erreurs.



Autre subtilité : le compilateur compile du texte brut et non pas enrichi comme peut en fournir Word (un texte enrichi est un texte auquel une mise en page a été ajoutée). Si jamais l'envie vous prend de copier-coller un texte depuis Word, faites d'abord un copier-coller dans le Bloc-notes ou un logiciel du même genre. Vous éviterez ainsi beaucoup de problèmes.

Compilation

Nous allons maintenant compiler votre fichier `.tex` pour le transformer en un autre format (PostScript, PDF ou DVI). Ici, trois cas de figure se présentent selon que vous soyez sous Windows, Mac ou Linux.

Compiler avec Kile (Linux)

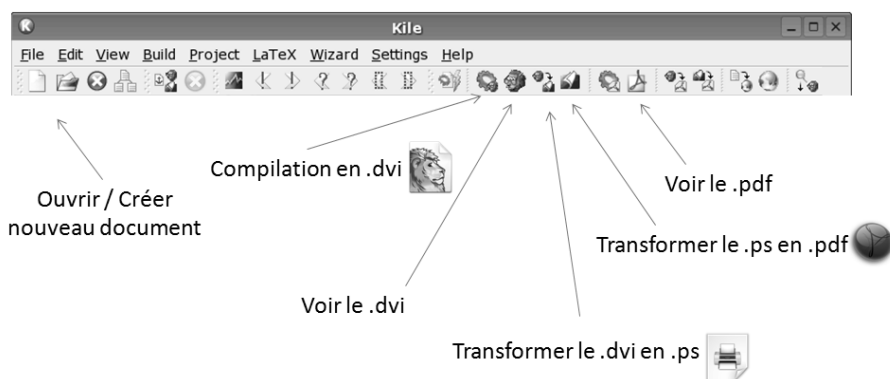


FIGURE 3.3 – Compilation avec Kile (pour les anciennes versions)

Vous pouvez voir sur la capture d'écran ci-dessus (figure 3.3) la barre d'outils à votre disposition dans Kile. Pour vous, tout commence par la création d'un fichier DVI que vous transformerez ensuite en fichier PostScript, puis en PDF selon vos besoins.

Pas de manipulation compliquée ici, votre éditeur $\text{\LaTeX}$ se charge de tout et les fichiers produits par la compilation se trouvent dans le dossier où vous avez enregistré votre fichier `.tex`.



J'ai choisi d'insérer ici des captures d'écran de l'ancienne version de Kile ; ce n'est pas une erreur de ma part², mais un choix pédagogique. La nouvelle version comporte simplement quelques menus déroulants extrêmement intuitifs³.

Compiler en ligne de commande (Linux)

Nous avons vu dans le chapitre précédent que l'éditeur $\text{\LaTeX}$ n'était pas indispensable. Il vous est possible de créer votre fichier `.tex` dans un éditeur de texte (Emacs, Vim. . .) avant de le compiler en ligne de commande.

2. Il est encore possible de travailler sur des ordinateurs utilisant l'ancienne version, même s'ils sont de plus en plus rares.

3. Ils sont du même type que ceux de $\text{\TeX}$ Shop.

La procédure est assez simple. Une fois votre fichier enregistré dans un dossier, ouvrez la console et placez-vous dans ce dossier avant d'utiliser l'une ou l'autre des commandes ci-après.

Dans un premier temps, vous pouvez compiler votre fichier `bonjour.tex` en un fichier `bonjour.dvi`. Le fichier créé apparaît alors dans le dossier où vous avez enregistré votre fichier d'origine en `.tex`.

```
latex bonjour.tex
```

Vous pouvez ensuite lire votre fichier `bonjour.dvi` grâce à la commande `xdvi` et l'imprimer grâce à la commande `dvips`.

```
xdvi bonjour.dvi  
dvips bonjour.dvi
```

Vous pourrez par la suite choisir de transformer votre fichier `bonjour.dvi` en un fichier `.ps` grâce à la commande `dvips` (en y ajoutant `-o` comme vous pourrez le voir plus bas dans l'exemple). La commande `ps2pdf` vous permettra de transformer votre fichier PostScript en un fichier PDF.

Il vous est tout à fait possible de compiler directement votre fichier `bonjour.tex` en un fichier `bonjour.pdf` grâce à la commande `pdflatex`. La lecture de PDF se fait grâce à la commande `xpdf`.

Résumons ces commandes par quelques exemples pratiques. Testez ces commandes, elles vous serviront tout au long du cours.

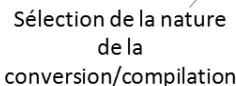
```
dvips bonjour.dvi -o  
ps2pdf bonjour.ps  
pdflatex bonjour.tex  
xpdf bonjour.pdf
```

Compiler avec T_EXShop (Mac)

Une fois de plus, c'est vous, détenteurs de Mac, qui possédez l'interface la plus minimaliste et la plus simple : il vous suffit de cliquer sur « Composition » tout en ayant sélectionné « LaTeX » dans le menu déroulant à droite du bouton.

Votre éditeur L^AT_EX se charge de tout et les conversions sont faites de façon automatique.

Sans plus attendre, vous pouvez admirer le résultat de votre compilation dans la fenêtre qui vient de s'ouvrir, ou en allant fouiller dans le dossier où est enregistré votre fichier `.tex`.



Visionneuse

FIGURE 3.4 – Compilation avec T_EXnicCenter

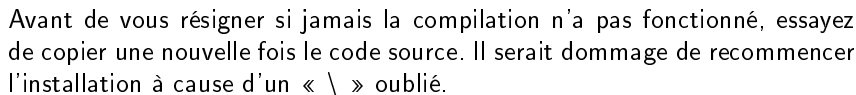
Compiler avec T_EXnicCenter (Windows)

Dans votre barre de menu, vous constatez la présence d'un menu déroulant (figure 3.4). Ce menu vous permet de choisir entre la compilation d'un fichier `.tex` en un `.dvi` et la conversion de ce fichier `.dvi` en un autre format de fichier.

Le bouton à sa droite sert à lancer les compilations / conversions et le bouton encore plus à droite vous donne les aperçus de vos fichiers. **Les fichiers créés apparaissent dans le dossier où vous avez préalablement enregistré votre .tex.**

Les résultats

Ici, deux possibilités s'offrent à vous : soit votre compilation a réussi, vous venez de découvrir comment écrire un document en L^AT_EX (libre à vous de l'imprimer) et vos logiciels fonctionnent, soit votre éditeur L^AT_EX vous a renvoyé une erreur... je suis alors au regret de vous annoncer que vous vous êtes trompé lors de l'installation et que vous devez la recommencer.



Vous venez de créer votre premier document, félicitations !

Types de documents et caractères spéciaux

Avant toute chose, sachez que la compilation de documents L^AT_EX fonctionne toujours de la même manière, vous n'avez plus rien à apprendre de ce côté-là.

Nous allons revenir ici sur le code que nous avons entré dans l'éditeur L^AT_EX et parler des caractères spéciaux.

Les caractères spéciaux

Comme tout langage de programmation, L^AT_EX utilise certains caractères pour son usage propre. Dans notre premier document, nous avons utilisé le « \ » pour lui envoyer des ordres. En somme, le « \ » n'est pas simplement recopié par L^AT_EX, mais interprété.

Ce n'est pas le seul caractère à avoir cette spécificité : ils sont dix et figurent dans le tableau ci-dessous.

Tentez de mettre l'un de ces caractères dans votre texte, il en résultera des erreurs de compilation. Pour signaler à L^AT_EX que nous ne lui envoyons pas une commande ou que nous ne communiquons pas avec lui en utilisant ces caractères, il suffit de faire précéder les signes d'un **backslash** (le fameux « \ »). Pour le **backslash** lui-même, nous utiliserons une commande.

Table des caractères réservés	
\$	\\$
&	\&
%	\%
#	\#
_	_
{	\{
}	\}
~	\~
^	\^
\	\textbackslash{}

Testons ce tableau ! Entrez le code suivant dans votre éditeur L^AT_EX, puis enregistrez le **.tex** avant de le compiler et de le convertir en PDF.

▷ Caractères spéciaux
Code web : 930209

```
\documentclass{article}

\begin{document}
Bonjour, je lis actuellement un cours traitant de LaTeX !
\$ \& \% \# \_ \{ \} \~ \^ \textbackslash{}

\end{document}
```

Vous devez obtenir la phrase suivante : « Bonjour, je lis actuellement un cours traitant de LaTeX! \$ & % # _ { } ~ ^ \ ».

Ce tableau est le premier, mais il y en aura d'autres, pour les accents par exemple. L'ensemble des tableaux de caractères spéciaux est rassemblé au sein d'une annexe volumineuse disponible à la fin du livre⁴.

4. Elle comporte surtout des caractères destinés aux mathématiques.

Types de documents et portions de texte

Nous allons ici décortiquer les trois lignes de code dont nous n'avons pas encore parlé.

Les portions de texte

Dans nos deux premiers exercices, nous avons entouré notre texte d'un couple de commandes.

```
| \begin{document}  
| \end{document}
```

Ces deux commandes délimitent le début et la fin de ce que nous appellerons un **environnement**. Ici, cela revient à dire que nous commencerons **toujours** un document par `\begin{document}` et que nous écrirons du texte avant de terminer le document par `\end{document}`.

De façon plus large, nous pourrions vouloir insérer un poème ou une recette dans un texte. Il nous suffirait alors d'écrire `\begin{recette}` puis la recette avant de fermer avec `\end{recette}`. Ce qui pourrait nous donner le code ci-dessous (ce n'est qu'un exemple fantaisiste).

```
| \begin{document}  
| \begin{recette}  
| Ma recette, bla, bla bla.  
| \end{recette}  
| \begin{texte}  
| Mon joli texte, bla, bla bla.  
| \end{texte}  
| \end{document}
```

La commande `\begin` ouvre un environnement tandis que `\end` le ferme. Le type d'environnement se voit entouré d'accolades et les options qui lui sont rattachées sont entre crochets (nous en utiliserons tout au long de ce livre).

En résumé, délimiter un environnement revient à écrire un code du même type que celui ci-dessous.

```
| \begin{type}[options]  
| Bla,bla  
| \end{type}
```

Nous apprendrons les noms et l'utilité des différents environnements au fil du cours.

Les types de documents

La première ligne de nos deux exercices n'a pas encore été analysée : elle contient pourtant une information capitale. Tout d'abord, réaffichons cette ligne, je vous expliquerai tout ensuite.

```
| \documentclass{article}
```

Une traduction approximative de la commande nous donne quelque chose qui ressemble à ceci : ce document est un article, probablement scientifique. Ici aussi, nous écrivons le type entre les accolades.

Cette commande sert en fait à expliquer à $\text{\LaTeX}$ que la publication que nous sommes en train de produire est un article. Ceci dans le but que $\text{\LaTeX}$ mette en page l'intégralité du contenu afin qu'il respecte les normes de typographie et de mise en page d'un article parfaitement présenté.



Pour écrire un roman, je fais comment ?

Tout simplement, vous remplacez le type de document par **book**. Il existe plusieurs types de documents, je vous cite ici les principaux dans un tableau.

Type de document	
article	Article (sous-entendu scientifique)
book	Livre
letter	Lettre
report	Rapport (stage, thèse...)



Peut-on là aussi spécifier des options ?

Absolument, nous verrons dans la suite du cours certaines options dédiées à certains types de document. La formulation des options se fait selon la syntaxe suivante :

```
| \documentclass[options]{type}
```

En résumé

- Pour transformer un fichier L^AT_EX (.tex) en PDF (.pdf), on passe par une étape dite de **compilation**.
- La compilation d'un document se déclenche grâce à des raccourcis présents dans l'éditeur L^AT_EX. Il est aussi possible d'utiliser la ligne de commande. Il suffit de taper : `pdflatex fichier.tex`.
- Certains caractères spéciaux (comme \$ # & %...) doivent être précédés d'un *backslash* (\) pour être insérés dans un texte⁵.
- Un document L^AT_EX peut être de type **article**, **book**, **letter** ou **report** selon le type de document que vous souhaitez écrire.
- Nous écrirons le contenu de notre document à l'intérieur de l'environnement **document**, c'est-à-dire entre les commandes `\begin{document}` et `\end{document}`.

5. L'oubli d'un *backslash* devant ces caractères spéciaux peut entraîner de multiples erreurs et bugs lors d'une compilation.

Chapitre 4

Les packages

Difficulté : 

Votre apprentissage des mécanismes élémentaires de $\text{\LaTeX}$ est terminé. Vous savez à présent ouvrir et rédiger des fichiers source en `.tex` et les compiler de façon à obtenir des documents aux formats PDF, PostScript et DVI.

Nous avons également pu constater que pour expliquer à $\text{\LaTeX}$ ce que nous attendons de lui, nous utilisons des commandes et des lignes de code. Ces différentes commandes et leurs applications seront étudiées dans les parties 2 et 3.

Ce chapitre est le dernier chapitre théorique. Nous allons ici répondre à une question importante : si $\text{\LaTeX}$ ne sait pas faire quelque chose (mettre du texte en couleur, par exemple), comment changer cela ?

Nous allons répondre à cette question en introduisant la notion de « package », notion centrale et omniprésente dans l'univers $\text{\LaTeX}$.



La philosophie des packages

Les habitués de l'informatique connaissent assez bien la signification des mots « **extension** », « **package** » et « **plugin** ». Ces éléments sont présents dans de nombreux programmes et dans cette partie, nous allons tenter de développer le concept de « **package** » pour ceux à qui il n'est pas familier.

Imaginons votre installation $\text{\LaTeX}$ comme une cuisine équipée. Vous pouvez cuire des aliments, poser des objets sur le plan de travail, etc. En bref, cette pièce peut vous rendre certains **services** tout comme $\text{\LaTeX}$ nativement vous permet d'écrire du texte simple (c'est ce que nous avons pu voir dans le chapitre précédent).

Vous vous préparez un plat et avez besoin d'un couteau pour le cuisiner, or ce couteau ne fait pas partie des **outils** fournis par la personne qui a conçu votre cuisine. C'est un problème. Il est nécessaire pour vous d'aller chercher quelque part cet outil, afin de l'apporter dans votre cuisine pour vous permettre d'atteindre votre objectif : découper ce que vous vouliez découper.

Dans le cas de $\text{\LaTeX}$, des outils ont été créés dans le but de pallier certains manques et d'implanter de nouvelles fonctions. Grâce à ces derniers, l'univers $\text{\LaTeX}$ permet de réaliser chaque jour de nouvelles choses. Ces outils sont appelés « packages », et tout un chacun peut en créer ainsi que les diffuser.

Si jamais un package vient à manquer, vous irez le chercher sur l'internet et l'utiliserez ensuite (gardez la métaphore du couteau en tête, **quand un outil manque, il suffit d'aller le chercher**¹).

En tant qu'utilisateur de $\text{\LaTeX}$, vous serez fréquemment amené à utiliser des packages. En résumé, deux possibilités s'offrent à vous lorsque vous souhaitez en manipuler un :

- **le package est déjà présent** dans votre installation $\text{\LaTeX}$ et il ne vous reste plus qu'à vous en servir (c'est comme si votre couteau était déjà dans un des tiroirs de votre cuisine) ;
- **le package est absent** de votre installation $\text{\LaTeX}$ et vous devrez aller le chercher et l'installer (comme si vous alliez au magasin chercher le couteau pour le mettre dans un tiroir).

Ces tâches sont très simples et nous allons apprendre dans la suite de ce chapitre à installer et utiliser les packages de $\text{\LaTeX}$.

Comment s'en servir ?

Même si nous allons ici apprendre à installer des packages, ce n'est pas l'objectif principal de ce chapitre. Des millions d'utilisateurs avant vous ont conçu des documents à l'aide de $\text{\LaTeX}$ et rendu certains packages incontournables. C'est la raison pour laquelle ils sont aujourd'hui souvent installés par défaut avec votre distribution $\text{\LaTeX}$

1. Dans le cas de logiciels de traitement de texte, cette logique peut vous apparaître comme une faiblesse. En effet, pourquoi ce logiciel n'embarque-t-il pas d'emblée toutes les fonctions qu'on attend de lui ? Tout simplement pour conserver la possibilité d'évoluer facilement via sa communauté.

(de la même manière que vous ne vous installeriez pas dans une maison sans meubles, les utilisateurs de $\text{\LaTeX}$ préfèrent installer dès le début les éléments essentiels leur permettant de travailler confortablement).

Cette partie va vous apprendre à manipuler les packages.

Pour utiliser un package avec $\text{\LaTeX}$, nous procéderons en deux étapes. Dans un premier temps, nous communiquerons à $\text{\LaTeX}$, au moyen d'une commande, que nous travaillerons avec tel ou tel package et dans un second temps, nous pourrons les utiliser dans nos publications.

La commande `\usepackage[option]{type}` permet d'appeler un package. Elle se place juste après la ligne dédiée à la commande `\documentclass`². Regardez l'exemple ci-dessous que nous allons commenter ensemble.

```
\documentclass{report}

\usepackage[latin1]{inputenc} % un package
\usepackage[T1]{fontenc}      % un second package
\usepackage[francais]{babel}  % un troisième package
\begin{document}

J'écris mon 3e document avec \og \LaTeX{} \fg.
\LaTeX{} est un langage créé par Leslie \bsc{Lamport}.

\end{document}
```



Ici, nous constatons que les annotations dans le code se font avec la syntaxe suivante : `% Mon commentaire`. Dans la suite du cours je me servirai des commentaires pour annoter des morceaux de code et vous aider à vous repérer.

N'hésitez pas à utiliser les commentaires dans votre travail pour mieux vous y retrouver.

Trois packages ont été utilisés dans l'exemple ci-dessus :

- **babel** est utilisé pour spécifier à $\text{\LaTeX}$ que vous écrivez en français ;
- **fontenc** et **inputenc** vous permettent d'utiliser tous les caractères de votre clavier.

2. La zone entre `\documentclass` et `\begin{document}` est le préambule du code. C'est la zone où nous placerons les appels de packages et diverses commandes. Il ne faut cependant pas confondre le préambule du code avec le préambule d'un livre, ce n'est pas du tout la même chose.



Petit test pour nos amis linuxiens. Créez avec votre bloc-notes un fichier contenant la phrase « Je maîtrise déjà les bases du LaTeX » avant de l'enregistrer sous le nom `codage.txt` et de taper dans votre console la ligne suivante : `file codage.txt`. Si la console inscrit ISO-8859, tout va bien. Si elle inscrit UTF-8³, vous devrez taper `\usepackage[utf8]{inputenc}` à la place de `\usepackage[latin1]{inputenc}` dans vos prochains documents.

Utilisateur sous Mac OS : il est important que vous enregistriez vos documents en Latin-1. TeXShop vous proposera dans le menu Préférences → Documents → Encodage de choisir l'encodage par défaut. Sélectionnez celui-ci : Occidental (ISO Latin 1).



`\up{}`, `\og` et `\fg` sont-elles de nouvelles commandes ?

Effectivement, ces commandes sont nouvelles. Le package **babel** permet l'utilisation de commandes propres à la langue choisie⁴. Entre autres, les guillemets français et les petites capitales⁵ en ce qui concerne le français. Ce package permet donc à **L^AT_EX** d'exécuter de nouvelles tâches et d'être plus efficace pour écrire en langue française.

Ci-dessous, un tableau vous montrant les commandes introduites par le package **babel**. N'hésitez pas à les essayer et à vous faire la main avec.

Package babel	
«	<code>\og</code>
»	<code>\fg</code>
Exposant	<code>\up{}</code>
Petites capitales	<code>\bsc{Lamport}</code>
1 ^o	<code>\primo</code>
2 ^o	<code>\secundo</code>
3 ^o	<code>\tertio</code>
4 ^o	<code>\quarto</code>
8 ^o (ou n'importe quel autre chiffre)	<code>\FrenchEnumerate{8}</code>
n ^o	<code>\no</code>
N ^o	<code>\No</code>
n ^{os}	<code>\nos</code>
N ^{os}	<code>\Nos</code>
40° (le caractère degré)	<code>40\degres</code>

3. UTF-8 et ISO-8859 désignent des processus d'encodage de caractères, c'est-à-dire la manière dont votre machine va coder les lettres. Bien que nous ne puissions pas voir directement la différence, se tromper entre UTF-8 et Latin-1 peut tout faire planter.

4. Nous n'utiliserons ici que les commandes de babel liées au français.

5. VOICI UN TEXTE EN PETITES CAPITALES.

Des centaines de packages existent, chacun rendant $\text{\LaTeX}$ plus performant. Ils vous permettent d'ajouter, à la carte, des fonctions à votre distribution. Souvent, lorsque vous avez envie de faire quelque chose d'original avec $\text{\LaTeX}$, quelqu'un a déjà fabriqué un package adapté⁶ et il ne vous reste plus qu'à l'installer puis à l'utiliser.

Comment installer un package ?

L'installation par défaut de $\text{\LaTeX}$ sur votre machine inclut déjà de nombreux packages. Il est cependant possible que vous ayez un jour besoin d'installer un package absent sur votre machine. Cette partie est là pour vous expliquer comment faire.

Notons que si vous avez installé $\text{\LaTeX}$ comme expliqué dans le chapitre 3, **il est peu probable qu'un package utilisé dans ce cours ne soit pas présent sur votre machine.**



Cette sous-partie n'étant utile que si vous tentez d'installer un package non présent dans votre distribution, vous pouvez ne pas la lire et passer au chapitre suivant sans crainte.

MiKTeX et son installation à la volée

Les utilisateurs de MiKTeX (Windows) ont de la chance en ce qui concerne les packages, car MiKTeX installe tout seul comme un grand les packages appelés lors de la compilation qui ne sont pas présents sur leur machine.

Si vous souhaitez savoir si un package se trouve dans votre installation MiKTeX, rien de plus simple, il vous suffit d'aller dans votre menu Démarrer et d'aller chercher le **Package Manager** (figure 4.1).

Chemin du Package Manager : MiKTeX → Maintenance → Package Manager.

Le Package Manager vous permettra de voir la liste des packages installés dans votre distribution (figure 4.2).

Autres distributions

Il existe plusieurs méthodes d'installation de package avec $\text{\LaTeX}$. J'ai sélectionné pour vous les deux plus faciles à mon sens. Elles devraient vous permettre d'utiliser la quasi-totalité des packages.

Les deux méthodes développées ici diffèrent légèrement, suivant que votre package est un fichier `.ins` ou `.sty`.

Dans de rares cas, les packages sont fournis sous d'autres extensions, mais ils sont alors accompagnés d'un fichier README vous guidant lors de leur installation.

6. Une recherche sur votre moteur de recherche favori suffit souvent à trouver la perle rare.

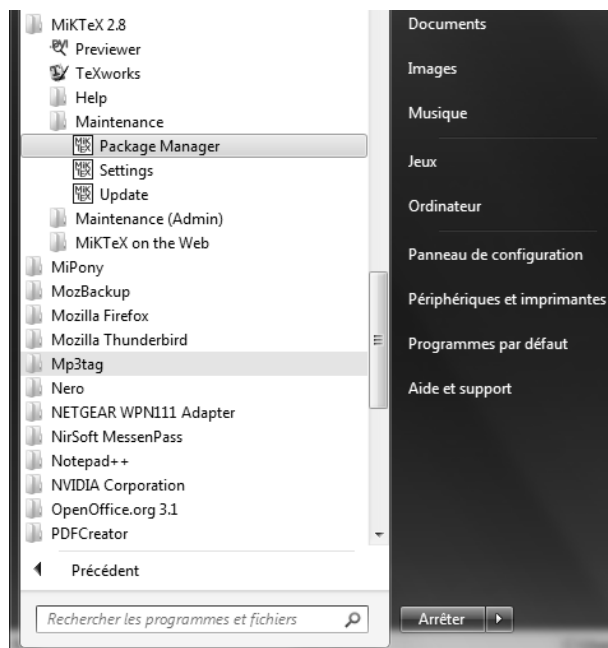


FIGURE 4.1 – Trouver le Package Manager dans le menu

Les packages en `.sty`, méthode simple

Si votre package est de la forme `nom_de_package.sty`, rien de plus simple pour l'utiliser : il suffit de le copier dans le dossier contenant votre source `.tex`. Lorsque votre distribution compilera le fichier `.tex`, elle recherchera dans ce dossier les fichiers `.sty` des packages manquants, et le tour sera joué.

Résumons, la commande `\usepackage{nom_de_package}` demande à L^AT_EX d'utiliser un package installé ou, s'il ne l'est pas, d'aller chercher le fichier `nom_de_package.sty` dans le dossier de travail.

Simple, n'est-ce pas ?

Les packages en `.ins`, méthode en deux temps

Les packages contenus dans un fichier `.ins` doivent être traités en deux étapes. Premièrement, mettez votre fichier `nom_de_package.ins` dans un répertoire et compilez-le : il enfantera (sans douleur) un fichier `nom_de_package.sty`.

Ce fichier `nom_de_package.sty` doit être traité selon le processus développé dans le paragraphe « Les packages en `.sty`, méthode simple ».

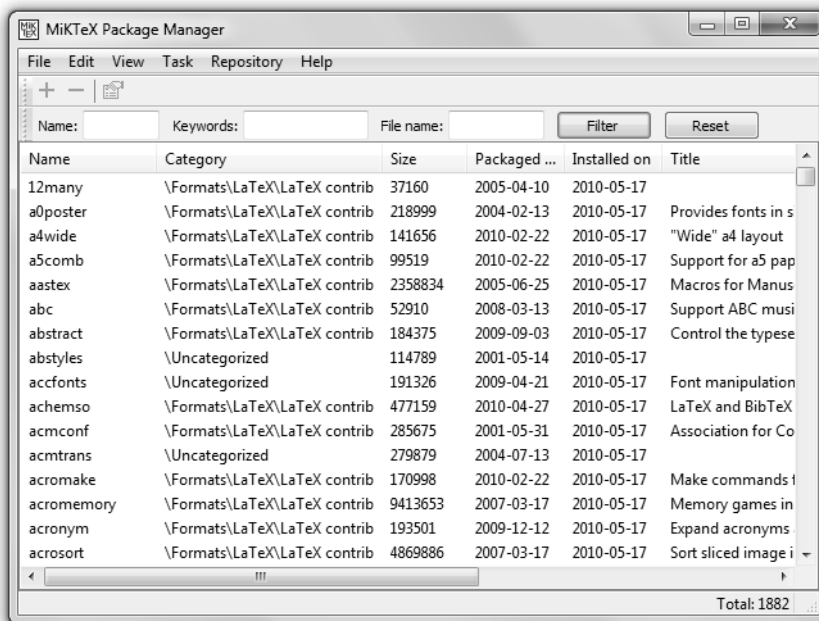


FIGURE 4.2 – Package Manager

En résumé

- Les packages sont des outils permettant à L^AT_EX d'exécuter de nouvelles tâches : coloration du texte, règles typographiques, lettrines, encadrements... Grâce à eux, il est possible d'étendre les possibilités de L^AT_EX.
- Il existe des milliers de packages. Sous Windows, MiK_TE_X les télécharge et les installe automatiquement à la volée. Sous Mac OS et Linux il faut parfois télécharger manuellement ces fichiers.
- Pour utiliser un package, il suffit de l'appeler au début de votre document avec la commande `\usepackage{nompackage}`.
- La communauté des utilisateurs (dont vous faites désormais partie) peut ajouter autant de fonctions qu'elle le souhaite à L^AT_EX via la création de packages ⁷.

7. Créez-en un qui prépare le café et vous ferez fortune. :-)

Deuxième partie

Utilisation basique de L^AT_EX

Chapitre 5

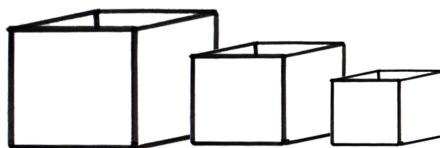
Maîtriser sa mise en page (1/2)

Difficulté : 

Les deux premiers chapitres de cette partie traiteront de mise en page. Nous y aborderons de nombreuses fonctions vous permettant de hiérarchiser votre contenu et de le présenter selon vos goûts.

Ce premier chapitre vous apprendra, entre autres, à :

- structurer vos documents ;
- maîtriser les différentes numérotations disponibles dans un document de classe book ;
- justifier et espacer vos paragraphes ;
- composer des préambules et annexes ;
- élaborer une page de garde.



Structure des documents

Généralités

Tableau des éléments de structure

Vous avez appris dans le chapitre 4 à indiquer à $\text{\LaTeX}$ que vous alliez écrire en français. Il est maintenant temps d'apprendre à hiérarchiser des informations sous forme de **parties**, **chapitres**, **sections**, etc. Votre distribution $\text{\LaTeX}$ se servira de la hiérarchisation de vos informations pour mettre en page votre contenu proprement ¹.

Ci-dessous, un tableau résumant les différents niveaux de hiérarchisation du contenu (de la grande partie jusqu'au tout petit sous-paragraphe).

Commandes des éléments de structure	
Partie	<code>\part{nom de la partie}</code>
Chapitre	<code>\chapter{nom du chapitre}</code>
Section	<code>\section{nom de la section}</code>
Sous-section	<code>\subsection{nom de la sous section}</code>
Sous-sous-section	<code>\subsubsection{nom de la sous sous section}</code>
Paragraphe	<code>\paragraph{nom du paragraphe}</code>
Sous-paragraphe	<code>\subparagraph{nom du sous paragraphe}</code>

Notez que `\chapter` n'est pas disponible pour les classes de document `article` et `letter`.

Exercice

J'ai choisi volontairement de vous présenter le tableau précédent avant de vous donner un exemple pratique. L'heure de l'exercice a sonné !

Essayez d'écrire un document de classe **report**, en français et contenant les éléments de structure suivants : une partie, un chapitre, une section et une sous-section dans laquelle vous écrirez quelques lignes.

Votre résultat doit ressembler à la figure 5.1 disponible à la page suivante.



Générateur de texte
Code web : 638427

Le texte latin utilisé pour cet exercice est un « faux texte », plus couramment appelé *lipsum*, généré sur l'internet. Ce type de texte permet de remplir un document d'informations factices afin de voir à quoi ressemblera le document final. C'est un outil précieux pour les webdesigners et les autres professionnels du graphisme et de la mise en page.

1. $\text{\LaTeX}$ se base sur différentes règles de typographie et de mise en page professionnelles pour donner à chaque élément un placement optimal et une lisibilité accrue.

Première partie Partie 1	Chapitre 1 Chapitre 1.1 Section 1.1.1 Une sous-section Lorum ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egetas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec nonum mauris, lacinia vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Place theclant orci eu velit varius luctus. Etiam laetitia viverra enim ac varius. Duis pretium elit eu erat auctor vel laetitia nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. 2
---	--

FIGURE 5.1 – Résultat de l'exercice

Si ceci est le premier document que vous composez par vous-même, il est normal que cela vous demande de rechercher des notions dans les chapitres précédents ; de même, il est normal de trouver quelques coquilles dans votre code. Néanmoins, votre code final doit être proche de celui-ci (même si vous pouvez avoir choisi des noms de chapitres ou de sections différents) :

```
\documentclass{report}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\begin{document}

\part{Partie}
\chapter{Chapitre}
\section{Section}
\subsection{Une sous-section}

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Donec nec condimentum libero. Phasellus condimentum porttitor congue.
Morbi eget quam sed justo egestas lobortis. Aenean et erat metus.
Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla.
Donec massa mau ris, luctus vitae auctor non, sagittis non sapien.
Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis.
Fusce tincidunt orci eu velit varius luctus.
Etiam iaculis viverra enim ac varius.
Duis pretium elit eu eros auctor vel iaculis nulla commodo.
Aliquam interdum fermentum orci sed fringilla.
Sed euismod condimentum dui, et pharetra ipsum dictum quis.

\end{document}
```

▷ Copier ce code
Code web : 725858

Modification de la numérotation des éléments de structure

Numéroter les annexes

Rendre un rapport nécessite souvent de mettre en annexe un certain nombre de textes et de photos². L^AT_EX permet à l’auteur de numéroter ses chapitres de contenu en chiffres et d’entamer une nouvelle numérotation des chapitres annexes avec des lettres.

La commande permettant de faire ceci est la commande `\appendix`. Elle se place entre le contenu et les annexes, comme vous pouvez le constater dans le prochain code source.

2. Cela permet, par exemple, de ne pas encombrer le contenu principal avec d’immenses tableaux.

```
\documentclass{report}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}

\begin{document}

\part{Partie}

\chapter{Premier chapitre}
\chapter{Second chapitre}
\chapter{Troisième chapitre}
\chapter{Quatrième chapitre}
\chapter{Cinquième chapitre}
\chapter{Sixième chapitre}
\appendix
\chapter{Un schéma}
% Pas besoin de changer de commande pour transformer un chapitre en annexe
% Grâce à appendix
\chapter{Un article annexe}
\chapter{Un canard}

\end{document}
```

▷ Copier ce code
Code web : 599182

Résumons ceci par un schéma (figure 5.2).

Dans un article, étant donné que la commande `\chapter` ne peut pas être utilisée (elle n'est utilisable que pour les rapports et les livres), `\appendix` jouera sur la numérotation des sections.



Il est possible de créer des chapitres sans numéro ni lettre en tapant `\chapter*{nom du chapitre}`. L'ajout d'étoile en fin de mot fonctionne avec tous les éléments de structure donnés plus haut. Vous pouvez donc choisir de créer des parties sans numéro. Il en va de même pour les paragraphes et les sections.

Autres numérotations particulières

Dans le cas d'un livre, il peut être intéressant d'organiser et de numérotter de façons différentes les pages suivantes :

- les préambules et avant-propos ;
- les chapitres de contenu ;
- les annexes ;

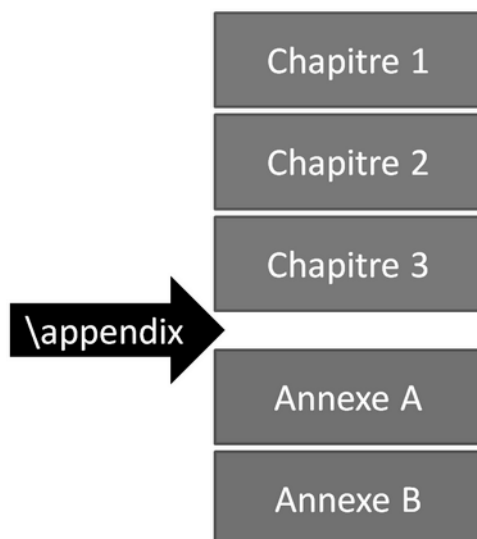


FIGURE 5.2 – Utilisation de la commande `\appendix`

– les pages d’index et de bibliographie³.

Ainsi, de nouvelles commandes apparaissent, fonctionnant de la même manière que la commande `\appendix`. Cependant, elles n’agissent pas uniquement sur la numérotation des chapitres et sections, mais également sur la numérotation des pages. Tout d’abord, la commande `\frontmatter`, qui se positionne juste après `\begin{document}`, permet de numérotter le préambule en chiffres romains.

Ensuite, `\mainmatter` se place entre le préambule et le premier chapitre. Cette commande permet de lancer la numérotation arabe habituelle des pages (1, 2, 3, etc.).

Enfin, `\backmatter` se place avant le chapitre épilogue, les index et bibliographies (que nous apprendrons à créer par la suite). Il a pour effet de stopper la numérotation des chapitres, mais pas la numérotation des pages.

La figure 5.3 résume tout ce qui a été dit.

Page de garde

La théorie

Nous venons de créer une organisation hiérarchique des données présentes dans votre document. Néanmoins, il manque une page extrêmement importante : la page de garde.

La page de garde est composée de trois éléments :

3. Nous apprendrons dans les chapitres 13 et 14 à créer des index ainsi que des bibliographies.

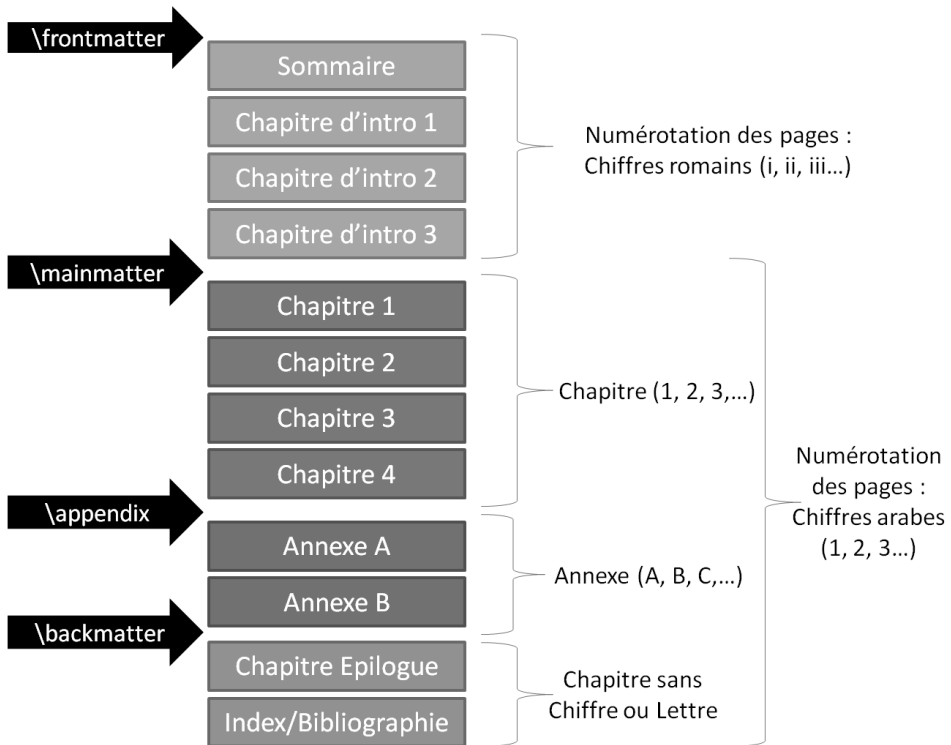


FIGURE 5.3 – Les autres éléments

- le titre du document (commande `\title{votre titre}`);
- l’auteur (commande `\author{les noms des auteurs}`);
- la date (commande `\date{la date que vous souhaitez}`).

Ces trois éléments sont introduits avant la commande `\begin{document}`, et une quatrième commande `\maketitle` se place juste après `\begin{document}` afin de faire comprendre à L^AT_EX que vous souhaitez composer une page de garde avec les trois éléments cités plus haut.

La pratique

La création d’une page de garde est assez simple et vous avez toutes les cartes en main pour en créer une. Vous allez donc effectuer un exercice dans lequel je souhaite que vous réalisiez un article avec une page de garde dans les conditions suivantes :

- l’auteur s’appelle John Doe;
- il a écrit le livre le 3 mai 2010;
- « Les Lapins » est le titre de son livre;
- le livre comporte une seule partie nommée « Éloge des lapins », qui comportera un paragraphe de faux texte.

Normalement, vous voici avec une belle page de garde (figure 5.4) et un code ressemblant, au paragraphe près, à celui cité ci-dessous :

```
\documentclass{book}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\title{Les Lapins}
\author{John \bsc{Doe}}
\date{3 mai 2010}
\begin{document}

\maketitle

\part{Éloge des lapins}

Votre gros paragraphe.

\end{document}
```



Copier ce code
Code web : 213003

Les Lapins

John DOE

3 mai 2010

FIGURE 5.4 – Page de garde

Alignements de texte et sauts

Alignements de texte

L^AT_EX justifie naturellement les paragraphes, il vous sera donc nécessaire d'utiliser différents environnements si vous souhaitez que votre texte soit centré, aligné à gauche ou à droite.



Rappel sur les environnements : un environnement commence par la commande `\begin{nom de l'environnement}` et se termine par `\end{nom de l'environnement}`.

Ici, trois environnements différents sont utilisables suivant vos besoins :

- l'environnement `flushright` pour aligner votre texte à droite ;
- l'environnement `center` pour centrer votre texte ;
- l'environnement `flushleft` pour aligner votre texte à gauche.

Une petite démonstration pratique de l'utilisation de ces trois environnements se trouve à la figure 5.5 (essayez de produire le même résultat : si vous n'y arrivez pas, reprenez le chapitre introduisant les environnements).

Sauts

Ici nous allons apprendre à sauter des lignes et créer des paragraphes. Rien de compliqué à cela, vous allez voir. Tout d'abord, **pour créer un paragraphe**, il vous suffit de sauter deux lignes. Rien de plus, juste deux sauts de lignes, comme vous pouvez le voir ci-dessous.

```
\begin{document}
Un paragraphe.

Un second paragraphe.
\end{document}
```

0.1 texte justifié

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut fringilla tempor diam, at dignissim enim pellentesque in. Phasellus mi arcu, fermentum euismod hendrerit sit amet, ultrices eu urna. Vestibulum congue velit id quam condimentum nec molestie nunc imperdiet. Maecenas ac mi ipsum, ut dictum nisi. In tincidunt vehicula dolor quis sagittis. Vestibulum nulla nibh, accumsan nec volutpat nec, faucibus a justo. Aenean aliquam aliquet rutrum. Suspendisse rhoncus sapien ac elit bibendum facilisis congue ante rutrum. Phasellus vestibulum porttitor luctus.

0.2 texte aligné à droite

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut fringilla tempor diam, at dignissim enim pellentesque in. Phasellus mi arcu, fermentum euismod hendrerit sit amet, ultrices eu urna. Vestibulum congue velit id quam condimentum nec molestie nunc imperdiet. Maecenas ac mi ipsum, ut dictum nisi. In tincidunt vehicula dolor quis sagittis. Vestibulum nulla nibh, accumsan nec volutpat nec, faucibus a justo. Aenean aliquam aliquet rutrum. Suspendisse rhoncus sapien ac elit bibendum facilisis congue ante rutrum. Phasellus vestibulum porttitor luctus.

0.3 texte centré

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut fringilla tempor diam, at dignissim enim pellentesque in. Phasellus mi arcu, fermentum euismod hendrerit sit amet, ultrices eu urna. Vestibulum congue velit id quam condimentum nec molestie nunc imperdiet. Maecenas ac mi ipsum, ut dictum nisi. In tincidunt vehicula dolor quis sagittis. Vestibulum nulla nibh, accumsan nec volutpat nec, faucibus a justo. Aenean aliquam aliquet rutrum. Suspendisse rhoncus sapien ac elit bibendum facilisis congue ante rutrum. Phasellus vestibulum porttitor luctus.

0.4 texte aligné à gauche

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut fringilla tempor diam, at dignissim enim pellentesque in. Phasellus mi arcu, fermentum euismod hendrerit sit amet, ultrices eu urna. Vestibulum congue velit id quam condimentum nec molestie nunc imperdiet. Maecenas ac mi ipsum, ut dictum nisi. In tincidunt vehicula dolor quis sagittis. Vestibulum nulla nibh, accumsan nec volutpat nec, faucibus a justo. Aenean aliquam aliquet rutrum. Suspendisse rhoncus sapien ac elit bibendum facilisis congue ante rutrum. Phasellus vestibulum porttitor luctus.

FIGURE 5.5 – Les différents alignements

Pour **aller à la ligne sans créer de nouveau paragraphe** il vous faudra taper la commande `\newline` ou la commande `\\`.

Pour **faire un saut de page** il vous faudra taper la commande `\newpage`. Cette commande ferme le paragraphe en cours et en crée un nouveau au début de la page suivante.

Ci-dessous (figure 5.6) un exemple d'utilisation de ces nouvelles commandes ainsi que son rendu.

| `Un canard\\un lapin\noline une pintade \newpage un ours`

Un canard un lapin une pintade 1	un ours 2
--	---------------------------------

FIGURE 5.6 – Rendu des différents types de sauts

En résumé

- Les documents $\text{\LaTeX}$ respectent une hiérarchie très précise : une partie contient des chapitres, scindés en sections, elles-mêmes divisées en sous-sections...
- Un document de classe `book` (livre) propose en plus un découpage global avec un préambule, un corps, des annexes et des chapitres épilogues⁴.
- Les différentes parties d'un livre n'ont pas la même numérotation, ni au niveau des titres, ni au niveau des numéros de pages.
- Une page de garde simple⁵ comporte un titre, le nom du ou des auteurs et une date.
- Les environnements `flushright`, `center` et `flushleft` permettent respectivement d'aligner à droite, de centrer ou d'aligner à gauche du texte. Par défaut, $\text{\LaTeX}$ justifie le texte.
- Pour créer un nouveau paragraphe, il suffit de sauter deux lignes. Si vous le souhaitez, il est aussi possible d'utiliser `\newline` ou `\\` qui permettent d'effectuer un simple retour à la ligne. Enfin, `\newpage` engendre un saut de page.

4. Dans cet ouvrage, nous utiliserons le terme « épilogues » pour les parties situées entre `\backmatter` et `\end{document}`, parce que $\text{\LaTeX}$ les numérote de manière différente et que cette particularité peut être utile à certains. Cependant, dans la plupart des ouvrages francophones, il n'y a bien entendu qu'un seul épilogue.

5. Il est bien sûr possible de faire une page de garde bien plus compliquée.

Chapitre 6

Maîtriser sa mise en page (2/2)

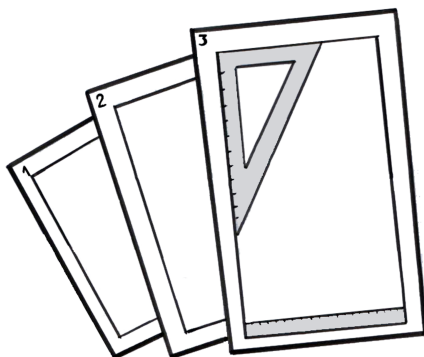
Difficulté : 

Dans ce second chapitre sur la mise en page, nous n'allons pas voir comment structurer les informations, mais plutôt apprendre à mieux les présenter.

Au programme de ce chapitre :

- la modification de la forme de votre document ;
- les marges ;
- la suppression de la numérotation de vos pages ;
- les listes numérotées et non numérotées ;
- les en-têtes et pieds de page.

Ces éléments sont cruciaux pour une bonne mise en page. Les marges vous permettront par exemple de relier vos documents plus facilement.



La forme de votre publication

La commande `\documentclass{}`

Nous avons vu dans la première partie qu'un document commence forcément par la commande `\documentclass[options]{type du document}`. Les types de documents utilisés dans ce livre sont énoncés dans le tableau ci-dessous (déjà présent dans le chapitre 3).

Type de document	
<code>article</code>	Article (sous-entendu scientifique)
<code>book</code>	Livre
<code>letter</code>	Lettre
<code>report</code>	Rapport (stage, thèse...)

Par le biais du type de document, nous avons défini la nature de la publication que nous composons : livre, lettre ou autre. Néanmoins, certaines informations manquent :

- le format du papier sur lequel sera imprimée la publication (A4, A5...);
- la taille de la police principale;
- l'alignement des équations (que nous apprendrons à écrire dans la partie 3);
- le positionnement de la première page de chaque chapitre (il est possible de faire en sorte que chaque chapitre commence sur la page de droite);
- l'information permettant de savoir si le document créé sera recto verso ou non.

Un second tableau fait donc naturellement son apparition, listant les options applicables à la commande `\documentclass{}`.

Options applicables à la commande <code>\documentclass{}</code>		
Description	Valeurs applicables	Valeur par défaut
Format du papier	<code>a4paper</code> , <code>a5paper</code> , <code>letterpaper</code> , <code>b5paper</code> ...	<code>letterpaper</code>
Taille de la police principale	<code>10pt</code> , <code>11pt</code> , <code>12pt</code>	<code>10pt</code>
Alignement des équations	<code>fleqn</code> (à gauche)	centrées par défaut
Colonnes	<code>onecolumn</code> , <code>twocolumn</code>	<code>onecolumn</code>
Première page des chapitres	<code>openany</code> , <code>openright</code>	<code>openright</code>
Recto verso	<code>oneside</code> , <code>twoside</code>	<code>article</code> et <code>report</code> : <code>oneside</code> , <code>book</code> : <code>twoside</code>

Exemples pratiques

Le tableau que vous venez de lire est un outil précieux, mais encore faut-il savoir s'en servir. Dans un premier temps, copiez et collez le code source ci-dessous dans un fichier `test.tex`.

```
\documentclass{book}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\begin{document}
\section{Un fort beau chapitre}
cinq ou six gros paragraphes de faux texte.
\end{document}
```

▷ Copier ce code
Code web : 619580

Tentez de remplacer `\documentclass{book}` par l'une des lignes ci-dessous (une seule à la fois : il ne faut pas deux commandes `\documentclass{}` dans une même publication).

```
\documentclass[twocolumn]{book}
\documentclass[twocolumn]{article}
\documentclass[12pt]{book}
\documentclass[11pt]{article}
\documentclass[twocolumn, 12pt]{book}
```

Essayez ces lignes (ajoutez un peu de texte pour mieux voir les différences, si vous préférez).

Retenez de cette sous-partie que la commande `\documentclass{}` peut être personnalisée grâce à des options contenues entre crochets et séparées par des virgules de la façon suivante : `\documentclass[option1, option2, option3]{type}`.

Marges et interlignes

Marges

Vous venez d'apprendre à créer des publications adaptées à la taille de votre papier et aux principales exigences d'un auteur (choisir la taille de la police principale est essentiel, par exemple).

Voyons maintenant comment modifier les marges d'un document.

Cette modification se fait en trois temps. Tout d'abord, nous allons créer un document dans lequel nous appelons le package `layout`, puis nous allons utiliser la commande `\layout` introduite par celui-ci afin d'obtenir un gabarit qui nous donnera une idée de la taille de chaque marge.

En parallèle, nous imprimerons un document saturé de texte afin de voir le rendu produit avec ces tailles de marges.

Le code à utiliser pour la création du `layout` est donné ci-dessous. Il est accompagné

d’une capture d’écran d’un document saturé en texte, ainsi que d’une capture d’écran d’un layout (figure 6.1).

```
\documentclass{article}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{layout}

\begin{document}

\layout

\end{document}
```

▶ Copier ce code
Code web : 801168

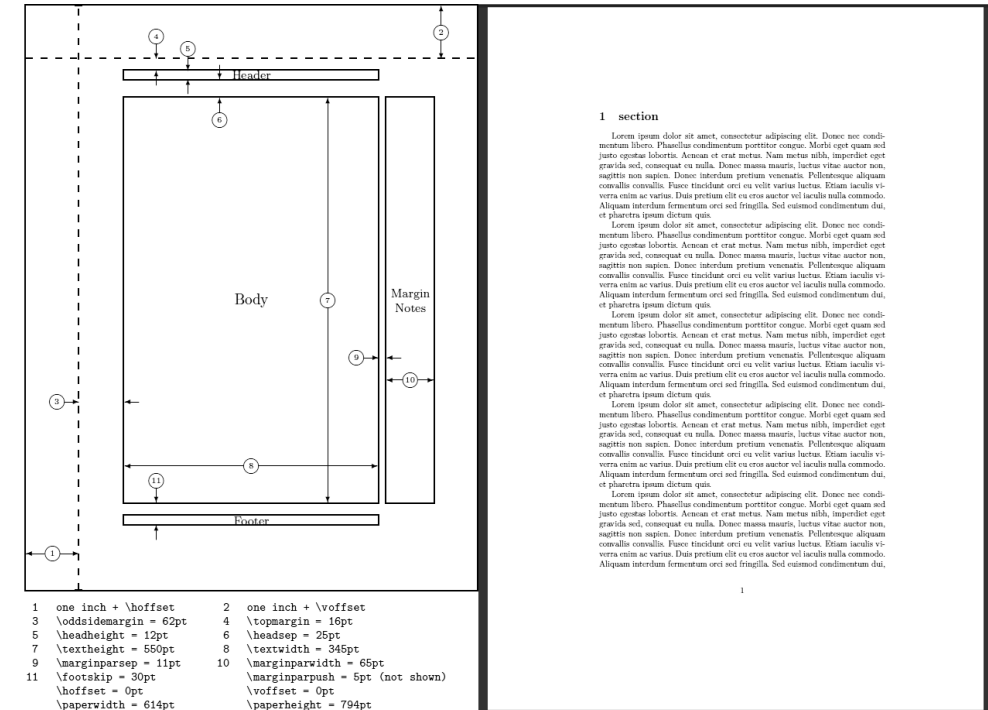


FIGURE 6.1 – Le duo layout / page saturée en texte

Ces deux éléments conjugués nous permettent de mieux visualiser les marges et leurs grandeurs respectives.

Afin de modifier celles-ci, nous utiliserons le package `geometry` dans lequel nous spécifierons les tailles des marges en haut, en bas, à gauche puis à droite.

Tout ceci s'écrit avec la syntaxe suivante :

```
| \usepackage[top=2cm, bottom=2cm, left=2cm, right=2cm]{geometry}
```

Bien sûr, vous pouvez modifier à votre guise les différentes longueurs.

Résumons les étapes :

- création du layout ;
- création d'une page **saturée** en texte (afin de mieux visualiser les marges) ;
- modification des marges (via le package `geometry`) ;
- création d'une page remplie de texte pour voir le rendu des marges (facultatif).

Interlignes

Dans certains types de documentation, un interlignage une fois et demie ou deux fois supérieur à la normale est requis. Nous allons voir ensemble comment utiliser ces derniers dans un document¹. Afin d'obtenir des interlignes personnalisés, il faut utiliser le package `setspace` ainsi que les commandes `\onehalfspacing` et `\doublespacing` dans le préambule, qui permettent d'obtenir dans tout le document un interligne respectivement 1,5 et 2 fois plus grand que l'interligne habituel.

Pour ne changer les interlignes que dans des petits morceaux de votre composition, les environnements `onehalfspace` et `doublespace` sont adaptés. L'exemple ci-dessous illustre ce cas (figure 6.2).

```
| \documentclass[10pt]{article}

| \usepackage[latin1]{inputenc}
| \usepackage[T1]{fontenc}
| \usepackage[francais]{babel}
| \usepackage{setspace}

| \begin{document}

| \section{interligne simple}

| Un paragraphe.

| \section{interligne intermédiaire}

| \begin{onehalfspace}
| Un autre paragraphe.
| \end{onehalfspace}
```

1. Il est également possible de créer des interlignes d'une taille définie par l'auteur, mais ce n'est pas notre souhait ici.

```
\section{interligne double}

\begin{doublespace}
Encore un autre paragraphe.
\end{doublespace}

\end{document}
```



Copier ce code
Code web : 763020

1 interligne simple

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus posuere vel ac metus. Aliquam pharetra mi in nibh molestie ac. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus.

2 interligne intermédiaire

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus posuere vel ac metus. Aliquam pharetra mi in nibh molestie ac. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus.

3 interligne double

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus posuere vel ac metus. Aliquam pharetra mi in nibh molestie ac. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed vitae est eget urna luctus.

FIGURE 6.2 – Interlignes

Les listes

Nous allons maintenant apprendre à créer des listes.

Les listes sont très utiles dans un document et peuvent être selon vos souhaits :

- des listes à puces (comme celle-ci) ;
- des listes numérotées ;
- des listes de description (beaucoup moins fréquentes).

Listes à puces

La création d'une liste à puces se fait en trois temps :

- ouverture de l'environnement `itemize`;
- écriture de chaque élément de la liste précédé de la commande `\item`;
- fermeture de l'environnement `itemize`.

Rien de compliqué là-dedans, vous pouvez voir ci-dessous un exemple pratique présentant un code de liste à puces, ainsi que son rendu (figure 6.3).

Petit bonus, le changement de la forme de la puce est introduit dans le code.

```
\begin{document}

\begin{itemize}

\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item[@] Une pintade. % En plaçant un @ entre crochets après \item,
↪ j'ai transformé la puce en @
\item[0] Un lapin.

\end{itemize}
```

▷ Copier ce code
Code web : 434978

Listes numérotées

La création d'une liste numérotée se fait également en trois temps (seul le nom de l'environnement change) :

- ouverture de l'environnement `enumerate`;
- écriture de chaque élément de la liste précédé de la commande `\item`;
- fermeture de l'environnement `enumerate`.

Si vous avez compris comment construire des listes à puces, vous n'aurez aucune difficulté à créer des listes numérotées.

Cependant, un exemple de code et un rendu en image (figure 6.4) ne peuvent pas faire de tort.

```
\begin{enumerate}

\item Un canard.
\item Un mammouth.
\item Un canard.
```

- Un canard.
- Un mammouth.
- Un canard.
- Un mammouth.
- Un canard.
- Un mammouth.
- @ Une pintade.
- 0 Un lapin.

FIGURE 6.3 – Liste à puces

```
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item Une pintade.

\end{enumerate}
```



Copier ce code
Code web : 436138

Listes de description

Si vous souhaitez écrire une série de définitions, l'environnement `description` est fait pour vous. Il vous permet de remplacer les puces par des expressions de votre choix qui seront mises en gras.

Ces mots en gras devront être placés entre crochets juste après la commande `\item` dans le code source.

Démonstration ci-dessous (figure 6.5) :

```
\begin{description}
```

1. Un canard.
2. Un mammouth.
3. Un canard.
4. Un mammouth.
5. Un canard.
6. Un mammouth.
7. Une pintade.

FIGURE 6.4 – Liste numérotée

```
\item[Un canard :] bestiole qui fait coin.
\item[Un poulpe :] bestiole qui fait bloub.
\item[Un ornithorynque :] bestiole qui fait rire.
\item[Un ours :] bestiole qui fait mal.

\end{description}
```

▷ Copier ce code
Code web : 838299

Un canard : bestiole qui fait coin.

Un poulpe : bestiole qui fait bloub.

Un ornithorynque : bestiole qui fait rire.

Un ours : bestiole qui fait mal.

FIGURE 6.5 – Liste de description

Les listes n'ont rien de sorcier : en faire usage aérera vos documents en plus de vous permettre, par exemple, de présenter simplement des processus, des manipulations ou des protocoles opératoires (ce ne sont que des exemples, vous pouvez très bien vous en servir pour lister des ingrédients dans la recette des crêpes).

Les styles

Nous venons d'apporter de nombreuses modifications à nos publications. Elles ont été mises en forme, structurées, listées. Il nous reste un élément essentiel à personnaliser : les **en-têtes** et **pieds de pages**.

Ces éléments ont eux aussi été étudiés par les concepteurs de L^AT_EX. Il existe bien sûr des packages permettant d'en faire des choses originales. Néanmoins, les pères de L^AT_EX ont choisi trois couples en-têtes / pieds de page qui leur semblaient les plus adaptés à un usage professionnel (nous utiliserons d'autres packages et des mises en forme élaborées plus loin dans le cours).

Pour utiliser un couple en-têtes / pieds de page, il vous suffira de choisir ce que L^AT_EX appelle un **style**. Ces trois styles sont :

- le style **plain** : il permet d'insérer le numéro de page au milieu du pied de page ;
- le style **headings** : il permet d'insérer le nom du chapitre et le numéro de page en en-tête. Le pied de page est vide ;
- le style **empty** : l'en-tête *et* le pied de page sont vides.

Dans les pages suivantes, vous trouverez des captures d'écran de documents composés respectivement avec les commandes **plain** (figure 6.6), **headings** (figure 6.7) et enfin **empty** (figure 6.8).

Pour donner un style à une page en particulier, il suffit d'utiliser la commande `\pagestyle{nom du style}` que nous insérerons au début de la page à modifier.

Rien de bien compliqué, n'est-ce pas ?

Dans le premier TP, nous apprendrons à personnaliser de manière bien plus approfondie nos en-têtes et pieds de page.

erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egestas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egestas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis.

erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egestas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egestas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis.

FIGURE 6.7 – Le style headings

erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec nec condimentum libero. Phasellus condimentum porttitor congue. Morbi eget quam sed justo egestas lobortis. Aenean et erat metus. Nam metus nibh, imperdiet eget gravida sed, consequat eu nulla. Donec massa mauris, luctus vitae auctor non, sagittis non sapien. Donec interdum pretium venenatis. Pellentesque aliquam convallis convallis. Fusce tincidunt orci eu velit varius luctus. Etiam iaculis viverra enim ac varius. Duis pretium elit eu eros auctor vel iaculis nulla commodo. Aliquam interdum fermentum orci sed fringilla. Sed euismod condimentum dui, et pharetra ipsum dictum quis.

FIGURE 6.8 – Le style empty

En résumé

- On peut personnaliser la forme de son document en ajoutant des paramètres à la commande `\documentclass`. Cela nous permet d'indiquer la taille du papier (A4, A5...), la taille de la police (10pt, 11pt...), le nombre de colonnes, etc.
- Il est possible de paramétrer les marges² d'un document. On fait appel pour cela au package `geometry`.
- Il existe différents types de listes, chacune adaptée au genre de données listées : liste normale (`itemize`), liste numérotée (`enumerate`), liste de description (`description`).
- La commande `\pagestyle` nous permet de choisir comment l'en-tête et le pied de page seront affichés : avec le numéro de page en pied de page (`plain`), avec le nom du chapitre et le numéro de page en en-tête (`headings`) ou sans en-tête ni pied de page (`empty`).

2. Ici, nous n'avons fait qu'un travail grossier, mais le premier TP vous apprendra à être beaucoup plus précis (en pratique vous constaterez que les tailles de marges par défaut sont souvent suffisantes).

Chapitre 7

Les polices

Difficulté : 

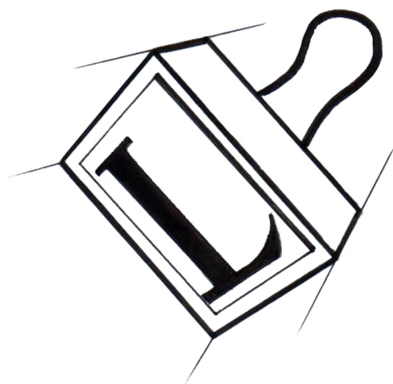
Ce troisième chapitre traitera des polices et de toutes les façons de modifier du texte. La bonne maîtrise de ces outils vous permettra de facilement mettre en avant des mots ou des phrases, rendant ainsi vos documents plus lisibles et structurés.

Quelques conseils seront distillés, en rapport avec les chapitres précédents (en particulier à propos de la commande `\documentclass` et de la taille de police par défaut).

À la fin de ce chapitre, vous pourrez sans problème :

- mettre en forme du texte (gras, italique, soulignement...);
- changer la couleur d'un texte;
- changer de façon ponctuelle ou définitive la police d'un document.

Rien de sorcier dans les commandes, leur manipulation ne devrait pas poser de problème. Ce chapitre étant le plus facile du livre, profitez-en pour vous entraîner.



Graisse, style, taille

Taille de texte

Nous allons dans ce paragraphe comprendre comment la taille du texte peut être modifiée, de façon à écrire certains mots en plus grand que les autres.

L^AT_EX propose dix commandes différentes permettant d'augmenter ou de diminuer la taille de votre texte selon votre envie. Ces commandes s'utilisent très simplement et possèdent deux syntaxes différentes :

```
\commande{mon bout de texte} % rien d'inhabituel  
{\commande mon bout de texte} % nouvelle façon
```

Dans le cas des modifications de tailles de texte, ces deux méthodes fonctionnent. Néanmoins, je vous conseille d'utiliser la syntaxe `\commande{mon bout de texte}` car elle est similaire à la majorité des autres syntaxes utilisées dans L^AT_EX.

Maman ?
Oui chéri !
il me faut...
de quoi as-tu besoin mon lapin ?
il me faut de l'eau !
Pardon ? !
De l'eau !
Il me faut de l'eau !
de l'eaaaaaaau !
de l'eaaauuuuuuuu !

FIGURE 7.1 – Les tailles de texte

Revenons à nos commandes. Comme dit plus haut, le texte peut subir dix transformations différentes au niveau de la taille. Cela nous donne donc une échelle de taille de texte assez large (figure 7.1).

Une fois n'est pas coutume, un beau tableau devrait nous aider à y voir plus clair dans ce tas de commandes. Il se trouve sur la page suivante¹.

1. Testez ces commandes, vous serez surpris des changements qu'elles effectueront dans vos mises en page.

Commande	Taille de texte
<code>\tiny</code>	Minuscule
<code>\scriptsize</code>	Très très petite
<code>\footnotesize</code>	Très petite
<code>\small</code>	Petite
<code>\normalsize</code>	Normale (définie dans <code>\documentclass</code>)
<code>\large</code>	Légèrement plus grande que la normale
<code>\Large</code>	Grande
<code>\LARGE</code>	Très grande
<code>\huge</code>	Très très grande
<code>\Huge</code>	Énorme!



Point important : lors de l'appel de `\documentclass`, vous avez choisi une taille de police pour votre document (10 pt par défaut, que vous pouvez augmenter jusqu'à 12 pt). Il serait maladroit d'encadrer l'intégralité d'un document dans une commande `\large` au lieu de simplement augmenter la taille de police par défaut.

Graisse, soulignement...

Nous venons de voir comment rendre plus petite ou plus grande une portion de texte. Nous allons à présent apprendre à mettre du texte en gras, à le souligner, ainsi qu'à lui appliquer moult commandes de mise en forme.

Il existe ici trois méthodes différentes pour appliquer des transformations de type graisse et soulignement à un texte :

- la méthode habituelle, `\commande{mon bout de texte}`;
- la méthode `{\commande mon bout de texte}`;
- les **environnements**.

À la différence des commandes permettant de modifier la taille du texte, s'utilisant indifféremment à l'intérieur ou à l'extérieur des accolades, il existe ici des commandes différentes suivant que vous utilisez l'une ou l'autre des deux méthodes.

Pour vous épargner de multiples paragraphes, les exemples et commandes sont synthétisés ci-dessous sous forme de tableau (à garder sous la main).



Gardez à l'esprit qu'un trop grand nombre de mots mis en forme peut nuire à votre document et le rendre totalement illisible.

Utilisez ces commandes avec parcimonie.

Mise en forme		
Modification	Commande	Rendu
Normal	<code>\normalfont un lapin</code> <code>\begin{rm}un lapin \end{rm}</code>	un lapin un lapin
Gras	<code>\textbf{un lapin}</code> <code>{\bfseries un lapin}</code> <code>\begin{bf}un lapin \end{bf}</code>	un lapin un lapin un lapin
Italique	<code>\textit{un lapin}</code> <code>{\itshape un lapin}</code> <code>\begin{it}un lapin \end{it}</code>	<i>un lapin</i> <i>un lapin</i> <i>un lapin</i>
Penché	<code>\textsl{un lapin}</code> <code>{\slshape un lapin}</code> <code>\begin{sl}un lapin \end{sl}</code>	<i>un lapin</i> <i>un lapin</i> <i>un lapin</i>
Machine à écrire	<code>\texttt{un lapin}</code> <code>{\ttfamily un lapin}</code> <code>\begin{tt}un lapin \end{tt}</code>	un lapin un lapin un lapin
Petites capitales	<code>\textsc{un lapin}</code> <code>\bsc{un lapin}</code> <code>{\scshape un lapin}</code> <code>\begin{sc}un lapin \end{sc}</code>	UN LAPIN UN LAPIN UN LAPIN UN LAPIN
Exposant (En mode texte)	<code>Un canardun lapin</code>	Un canard ^{un lapin}
Encadrer (Paramétrable)	<code>\fbox{un lapin}</code>	un lapin
Soulignement (Package soul)	<code>\ul{un lapin}</code>	<u>un lapin</u>
Soulignement (Package ulem)	<code>\uuline{un lapin}</code>	<u>un lapin</u>
Soulignement (Package ulem)	<code>\uwave{un lapin}</code>	<u>un lapin</u>
Barrer (Package soul)	<code>\st{un lapin}</code>	un lapin

La commande `\emph`, un cas à part

Nous venons de voir une vingtaine de commandes ayant chacune un usage bien défini. Certaines grossissent le texte, d'autres le barrent ou le soulignent. En bref, chacune a une fonction simple et facile à comprendre. La commande `\emph{texte}` est différente et permet tout simplement de dire à L^AT_EX « *ce bout de texte est important, mets-le en valeur* ». L^AT_EX se chargera alors tout seul de choisir la meilleure façon de le mettre en valeur.

Petit exemple, un mot mis en valeur dans un texte en italique, et un mot mis en valeur dans un texte normal (figure 7.2).

```
\documentclass{book}
\begin{document}
\textit{Lorem ipsum dolor sit amet, \emph{consectetuer} adipiscing elit.}
Lorem ipsum dolor sit amet, \emph{consectetuer} adipiscing elit.
\end{document}
```

▷ Copier ce code
Code web : 972530

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.
Lorem ipsum dolor sit amet, *consectetuer* adipiscing elit.

FIGURE 7.2 – La commande `\emph`

Vous constatez que `\emph` change de comportement suivant le contexte ; en cela réside sa force.

Conclusion : utiliser l’italique peut souvent être une erreur, mieux vaut lui préférer `\emph`. Dans le premier TP, nous apprendrons à faire en sorte que `\emph` mette en gras les mots importants.

Couleur

Nous allons ici apprendre à colorer du texte et à le rendre tel un arc-en-ciel au cœur de vos publications. Soulignons tout de même la dangerosité avérée d’une portion de texte rose au sein d’une thèse et la probabilité élevée que cette portion cause pertes et fracas à l’heure du jury.

Les huit couleurs par défaut

Pour la colorisation du texte, nous utiliserons le package `color`.

Ce package propose par défaut huit couleurs (**black**, **white**, **red**, **green**, **blue**, **yellow**, **magenta** et **cyan**), que vous pourrez utiliser à votre guise pour colorer du texte via la commande suivante :

```
| \textcolor{couleur}{texte en couleur}
```

Création de nouvelles couleurs

Avec huit couleurs, vous ne pouvez guère aller loin, raison pour laquelle de bonnes âmes ont conçu un système vous permettant d'en composer de nouvelles.

Pour ce faire, une commande est disponible : `\definecolor`.

Cette commande permet de créer des couleurs à partir de niveaux de gris ou d'un mélange de trois couleurs (rouge, vert et bleu).

Les couleurs créées recevront chacune un nom que vous choisirez et s'utiliseront de la même manière que précédemment, toujours grâce à la commande `\textcolor` et selon la syntaxe vue plus haut :

```
| \textcolor{le nom de couleur choisi}{texte en couleur}
```

Niveaux de gris

Définir une couleur en niveaux de gris revient à choisir une couleur entre le blanc et le noir. Ici, nous estimerons que le gris se trouve sur une échelle située entre 0 (le noir) et 1 (le blanc). Choisir un niveau de gris consistera à prendre un nombre à deux décimales (nous utiliserons un point entre le 0 et ses décimales) situé entre 0 et 1.

La syntaxe de `\definecolor` dans le cas des nuances de gris sera :

```
| \definecolor{le nom choisi sans espace}{gray}{le nombre à deux décimales}
```

Mélange de rouge, vert et bleu

Afin de créer de nouvelles couleurs, nous mélangerons trois couleurs initiales dans des proportions choisies. De la même manière que pour les nuances de gris, nous choisirons tour à tour l'intensité de rouge, de vert et de bleu que contiendra la couleur que nous voulons créer.

La syntaxe de `\definecolor` dans le cas du mélange rouge, vert, bleu sera :

```
| \definecolor{le nom de couleur choisi}{rgb}{taux de rouge entre 0 et 1,taux de  
| vert,taux de bleu}
```

Les packs de polices

Nous avons appris à changer la taille d'un morceau de texte et à lui faire subir de multiples transformations, de l'italique au soulignement. C'est bien, mais il nous reste encore une chose à apprendre à changer : **les polices de caractères**.

Il y a fort à parier que depuis le début de cet ouvrage, au fur et à mesure de vos compilations, l'envie vous a titillé de changer la police de vos textes.

Pour ce qui est des polices, L^AT_EX a des avantages à double tranchant. Rappelons-nous que L^AT_EX a été développé suite à l'exaspération de chercheurs face à la mise en page plutôt passable de leurs textes. Ils ont souhaité lors de la création du langage que l'ensemble des éléments d'une publication, qu'ils soient en gras, en majuscules ou dans une formule, soient cohérents entre eux.

Cette réflexion profonde a entraîné une innovation : la création des packs de polices.

L'idée centrale des packs de polices est de donner une cohérence au niveau de la typographie à l'intégralité du texte présent dans un document. Résultat : un pack comprenant quatre polices cohérentes correspondant respectivement à :

- des caractères avec empattements ;
- des caractères sans empattements ;
- des caractères façon machine à écrire (à chasse fixe) ;
- des caractères servant à écrire des formules mathématiques.

Bien qu'il existe un pack par défaut fourni dans L^AT_EX (utilisant la police **Computer Modern**) vous pouvez changer de pack de polices via l'utilisation des packages cités ci-dessous.

Il vous suffira de les appeler via la commande `\usepackage{nom_du_pack}`.

Une fois n'est pas coutume, voici une longue liste de noms avec les captures d'écran correspondantes.

Texte avec empattements

Texte sans empattements

Texte type machine à écrire

$$\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$$

FIGURE 7.3 – Par défaut

Des modifications ponctuelles de police peuvent aussi être introduites grâce à la commande suivante :

```
|{\fontfamily{code_de_la_police}\selectfont mon bout de texte}
```

Texte avec empattements
Texte sans empattements
Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.4 – Pack bookman

Texte avec empattements
Texte sans empattements
Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.5 – Pack charter

Texte avec empattements
Texte sans empattements
Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.6 – Pack newcent

Texte avec empattements
 Texte sans empattements
 Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.7 – Pack lmodern (le classique)

Texte avec empattements
 Texte sans empattements
 Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.8 – Pack mathpazo

Texte avec empattements
 Texte sans empattements
 Texte type machine à écrire
 $\int_4^8 kx dx \sum_{i=0}^n y_n^7 + 4569852321$

FIGURE 7.9 – Pack mathptmx

Différentes polices sont utilisables, j'en ai sélectionné pour vous et vous les livre dans un nouveau tableau (toujours avec des exemples de rendu).

Code de la police	Exemple
bch	Charter
cmr	Computer Modern
lmr	Latin Modern Roman
lmss	Latin Modern Sans Empattement
lmssq	Latin Modern Sans Emp. Exp.
lmtt	Latin Modern Typewriter
pag	Avant Garde
pcr	Courier
pbk	Bookman
phv	Helvetica
pnc	New Century Schoolbook
ppl	Palatino
ptm	Times
pzc	<i>Zapf Chancery</i>



Il est souvent tentant de changer en permanence de police. Néanmoins, un document sobre sera bien plus lisible pour un correcteur ou un lecteur qu'un document nécessitant en permanence de se réadapter à la police. Utilisez ces commandes sans lourdeur, sans quoi vos publications deviendront indigestes.

En résumé

- La taille du texte peut être paramétrée avec des commandes comme `\small`, `\normal`, `\large`, `\huge`...
- De nombreuses mises en forme sont utilisables via les commandes appropriées. Ainsi, il est notamment possible de passer le texte en souligné (`\u1`), italique (`\textit`), gras (`\textbf`)...
- Les créateurs de L^AT_EX ont privilégié des **packs** cohérents typographiquement, aux polices indépendantes. Il est possible de changer de pack de polices : il faut charger le package correspondant au pack choisi.

Chapitre 8

Les notes

Difficulté : 

Dans ce chapitre, les différents types de notes et de références possibles avec $\text{\LaTeX}$ seront abordés. C'est un chapitre facile, mais important, car il vous permettra de citer proprement tout ou une partie d'un texte, et de créer des notes en bas de certaines de vos pages.

Vous apprendrez à :

- citer des textes ;
- insérer des paragraphes de code brut dans vos documents ;
- créer des minipages ;
- composer des notes de bas de page ;
- utiliser les références internes au sein de votre publication.



Citation, code brut et URL

Cette première partie développe simplement les manières de créer des citations de texte, de code ainsi que la façon d'obtenir une URL cliquable.

Les citations simples

Fréquemment, vos rapports nécessiteront que vous citiez des propos. Pour vous permettre cela, deux environnements vous sont proposés : `quote` et `quotation`.

Ce code source vous montre la différence entre `quote` et `quotation` : le nombre de tabulations avant le premier mot ¹.

```

| Lorem ipsum dolor sit amet, consectetur adipiscing elit.
| Fusce condimentum tempor risus cursus aliquet.
| \begin{quote}
| Tout individu a droit à la vie, à la liberté et à la sûreté
| ↪ de sa personne. \end{quote}
| consectetur adipiscing elit.
| Lorem ipsum dolor sit amet, consectetur adipiscing elit.
| Nulla volutpat metus ut magna aliquam nec porttitor neque sodales.
| \begin{quotation}
| Tout individu a droit à la vie, à la liberté et à la sûreté
| ↪ de sa personne. \end{quotation}
| Lorem ipsum dolor sit amet, consectetur adipiscing elit.
| Fusce condimentum tempor risus cursus aliquet.
| Lorem ipsum dolor sit amet, consectetur adipiscing elit.
```

Lorem ipsum dolor sit amet, consectetur adipiscing
 elit. Fusce condimentum tempor risus cursus aliquet.

Tout individu a droit à la vie, à la liberté
 et à la sûreté de sa personne.

consectetur adipiscing elit. Lorem ipsum dolor sit
 amet, consectetur adipiscing elit. Nulla volutpat me-
 tus ut magna aliquam nec porttitor neque sodales.

Tout individu a droit à la vie, à la
 liberté et à la sûreté de sa personne.

Lorem ipsum dolor sit amet, consectetur adipiscing
 elit. Fusce condimentum tempor risus cursus aliquet.
 Lorem ipsum dolor sit amet, consectetur adipiscing
 elit.

1. `quotation` est fait pour de gros volumes de texte, tandis que `quote` est plus adapté pour une ligne isolée.

Les citations de code

Trois méthodes majeures permettent de citer des extraits de code.

La commande `\verb`

Pour insérer du code dans un paragraphe, le plus simple reste d'utiliser la commande `\verb`. Le texte que mettra en forme la commande `\verb` sera délimité par le caractère venant juste après son appel, généralement `|`.

```
| \verb| Mon bout de code {} |
```

Il se peut que vous utilisiez le caractère `|` dans certaines zones de votre code, et vous allez être confronté au problème que montrent le code source et le paragraphe ci-dessous (lisez-les attentivement).

```
| \verb[lapin & lapin {}] [  
| \verb|lapin & lapin {} |  
| \verb(lapin & lapin {} (  
| \verb&lapin & lapin {} &
```

Les trois premières lignes de code donnent le même résultat tandis que la quatrième ne fonctionne pas ; le `&` est le délimiteur dans la quatrième ligne et, étant donné qu'il y a déjà un `&` dans le code, $\text{\LaTeX}$ comprendra mal l'instruction et n'encadrera que le premier mot, « lapin ». Cette subtilité rend assez risquée l'utilisation de `\verb` et peut parfois déclencher des dizaines d'erreurs ou de mauvais rendus.

L'environnement `verbatim`(`tab`)

L'environnement `verbatim`, accompagné du package du même nom, permet d'encadrer de gros volumes de code. Petit souci : il remplace les tabulations par des espaces.

```
| \documentclass{article}  
  
| \usepackage[latin1]{inputenc}  
| \usepackage[T1]{fontenc}  
| \usepackage[français]{babel}  
| \usepackage{verbatim}  
| \begin{document}  
  
| \begin{verbatim}  
| votre code  
| \end{verbatim}  
| \end{document}
```

▷ Copier ce code
Code web : 522168

Afin d'éviter ce désagrément et de choisir librement le nombre d'espaces correspondant à une tabulation, vous pouvez utiliser l'environnement `verbatimtab` fourni par le package supplémentaire `moreverb`.

Sa syntaxe est la suivante :

```
\begin{verbatimtab}[nombre_d'espaces_par_tabulation]
votre code
\end{verbatimtab}
```

Dans l'exemple ci-dessous, chaque tabulation introduite dans le code se verra remplacée par 10 espaces une fois le document compilé.

```
\documentclass{article}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{verbatim}
\usepackage{moreverb}
\begin{document}

\begin{verbatimtab}[10]
votre code
\end{verbatimtab}
\end{document}
```

▷ Copier ce code
Code web : 376482

L'environnement `lstlisting`

L'environnement `lstlisting` permet de mettre en forme de façon colorée et d'utiliser de nombreuses options pour afficher du code.

Dans un premier temps, vous devez appeler le package `listings` dans l'en-tête du document, avant d'utiliser la commande `\lstset` et de la paramétrer pour qu'elle mette en forme votre code selon vos souhaits.

La commande `\lstset` se place avant `\begin{document}` et possède un grand nombre d'arguments paramétrables.

```
\lstset{ %
language=nom_du_langage,           % choix du langage
basicstyle=\footnotesize,          % taille de la police du code
numbers=left,                       % placer le numéro de chaque ligne à gauche (left)
numbers=right,                      % placer le numéro de chaque ligne à droite (right)
numberstyle=\normalsize,           % taille de la police des numéros
numbersep=7pt,                     % distance entre le code et sa numérotation
```

```
backgroundcolor=\color{white}, % couleur du fond
% Possibilité d'utilisation du package color
}
```

Les langages compatibles avec la commande sont constamment mis à jour sur la page de Wikibooks consacrée au package `listings`.

▷ Langages connus de listings
Code web : 491368

Jetez-y un coup d'œil à chaque fois que vous utiliserez ce package.

Le code à afficher s'insère dans votre document via l'environnement `lstlisting`.

Ci-dessous, un petit exemple du traitement d'un code C (extrait du tutoriel C du Site du Zéro) avec `lstlisting` (figure 8.1).

```
\documentclass{article}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{listings}
\lstset{
  language=C,
  basicstyle=\footnotesize,
  numbers=left,
  numberstyle=\normalsize,
  numbersep=7pt,
}

\begin{document}

\begin{lstlisting}
if (age == 2)
{
    printf("Salut bebe !");
}
else if (age == 6)
{
    printf("Salut gamin !");
}
else if (age == 12)
{
    printf("Salut jeune !");
}
\end{lstlisting}

\end{document}
```

▷ Copier ce code
Code web : 363592

```
1  if ( age == 2)
2  {
3      printf("Salut_bebe!");
4  }
5  else if ( age == 6)
6  {
7      printf("Salut_gamin!");
8  }
9  else if ( age == 12)
10 {
11     printf("Salut_jeune!");
12 }
```

FIGURE 8.1 – Citation de code

Gestion des URL

L^AT_EX gère les URL de façon très efficace, vous permettant de rendre vos documents interactifs.

Pour insérer une URL, il vous suffit d'utiliser le package `url` ainsi que la commande `\url{adresse}`, rien de plus compliqué que cela.

Minipage et texte encadré

Il peut parfois être utile de faire ressortir une portion de texte de façon singulière. L'environnement `minipage` et la commande `\fbox` vous permettent d'encadrer du texte et de le mettre en valeur.

Répetons-le une fois de plus, ce n'est pas parce que vous saurez faire des cadres après ce paragraphe qu'il faut en mettre partout.

Restez sobre, vos lecteurs vous en remercieront.

La commande `\fbox`

La commande `\fbox`, permettant d'encadrer des portions de texte, a déjà été vue dans le chapitre précédent.

Cependant, nous n'avons pas encore vu les paramètres qui peuvent lui être appliqués. `\fbox` est une commande puissante, pour laquelle il est possible de paramétrer des tas de choses. Néanmoins, dans ce cours, nous n'utiliserons `\fbox` que pour encadrer des textes.

Les deux paramètres qui nous intéressent ici sont l'écart entre le texte et la bordure ainsi que l'épaisseur de cette dernière. Ci-dessous une démonstration de `\fbox` et son résultat après compilation (figure 8.2).

```
% Commande permettant de définir l'écart
\setlength{\fboxsep}{8mm}
% Commande permettant de définir l'épaisseur du trait
\setlength{\fboxrule}{2mm}
\fbox{Un lapin}
```



FIGURE 8.2 – Texte encadré

L'environnement minipage

Une minipage est un encart de texte de largeur choisie, en quelque sorte une nouvelle page dans votre page². À l'intérieur de cet encart de texte, vous pourrez disposer et utiliser des environnements comme si cette minipage était un document à part entière.

Une minipage est en quelque sorte un document dans un document. Il vous est même possible d'insérer une minipage dans une minipage (mais nous allons éviter de faire des choses qui ne servent à rien :p).

L'environnement `minipage` est dépendant de deux paramètres : **la largeur** et **l'alignement vertical** de la minipage à créer.

Pour ce qui est de la largeur, je pense que vous avez deviné à quoi servait le paramètre correspondant³. Le paramètre d'alignement est quant à lui plus difficile à comprendre. L'alignement est vertical et défini par rapport au niveau du texte. Trois alignements sont proposés, et un schéma vaut mieux qu'un long discours pour visualiser les trois alignements disponibles.

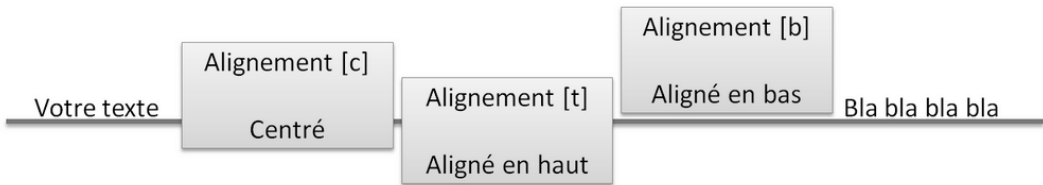


FIGURE 8.3 – Alignements possibles des minipages

Ici, une démonstration d'une minipage⁴ de 5 centimètres de large centrée verticalement

2. Des utilisateurs avancés de $\text{\LaTeX}$ peuvent réussir à agencer les minipages de manière surprenante, tout comme les boîtes encadrées.

3. Il s'exprime généralement en centimètres.

4. Elle est volontairement mise en plein milieu d'un long texte.

(figure 8.4) obtenue via la commande :

```
\fbox{ %fbox est utilisé pour voir les bords de la minipage
\begin{minipage}[c]{5cm}
Un petit paragraphe.
\end{minipage}
}
```

Aenean eu tristique lectus. Aenean velit lorem, aliquam vitae mattis eget, imperdiet eu lorem. Aenean posuere sollicitudin hendrerit. Nullam sed adipiscing neque. Nulla sapien justo, feugiat et ultricies at, volutpat at nibh. Mauris placerat ornare urna, sit amet dapibus libero bibendum vitae. Nunc venenatis tristique augue non semper. Donec et ultricies ante. Nulla facilisi. Mauris eu urna

et augue dictum consequat in vitae nibh.

Suspendisse potenti. Sed quis nisi scelerisque eros posuere tempor. Aliquam euismod tristique odio, in congue diam condimentum vitae. Nulla facilisi.

Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut libero risus, ultrices non imperdiet vel, sodales ut metus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed placerat mollis est, ut dapibus eros mattis at. Suspendisse nisi diam, rhoncus sit amet fringilla eu, ultricies sit amet nunc. Aenean sit amet mauris elit. Nunc ma-

FIGURE 8.4 – Exemple de minipage

Notes de bas de page, références internes

Cette partie traite des notes de bas de page ainsi que des références internes au sein des documents.

Ces deux éléments, souvent négligés, montrent à votre lectorat que votre texte est structuré et bien pensé.

Notes de bas de page

Il existe deux écoles pour l'insertion de notes de bas de page.

La commande `\footnote`

La première consiste à utiliser la commande `\footnote{Texte de la note.}` tout simplement à l'endroit où vous voulez insérer un renvoi vers une note de bas de page.

Cette commande est très intuitive, la seule chose à paramétrer ici est l'espacement entre l'exposant renvoyant à la note et le mot ou le groupe de mots concerné. Trois solutions sont possibles :

- coller l'exposant au mot concerné;
- utiliser une espace⁵, qui peut parfois être trop large;
- insérer une espace plus fine via la commande `\espace`.

```
un canard\footnote{bestiole qui fait coin}
un ornithorynque\footnote{bestiole qui fait rire}
un ours\footnote{bestiole qui fait mal}
```

un canard¹ un ornithorynque² un ours³

-
1. bestiole qui fait coin
 2. bestiole qui fait rire
 3. bestiole qui fait mal

FIGURE 8.5 – Notes de bas de page (méthode simple)

La commande `\footnotemark`

Ici, l'opération se déroule en deux temps. Dans un premier temps, vous marquerez tous les éléments concernés par des notes de bas de page personnalisées via la commande `\footnotemark[n'_de_la_note]`, puis vous indiquerez les notes de bas de page correspondant aux numéros précédemment choisis grâce à la commande `\footnotetext[n'_de_la_note]{Texte}`.

Deux compilations sont nécessaires : une pour que $\text{\LaTeX}$ liste les marques, et une pour qu'il associe les notes de bas de page aux marques. Cette méthode est souvent salvatrice lorsque des erreurs apparaissent avec `\footnote`. Ci-dessous, un code source de démonstration ainsi qu'une capture d'écran prise après deux compilations.

```
un canard\footnotemark[1] \\\n ornithorynque\footnotemark[18] \\\n ours\footnotemark[32]\n\n\footnotetext[1]{bestiole qui fait coin}\n\footnotetext[18]{bestiole qui fait rire}\n\footnotetext[32]{bestiole qui fait mal}
```

5. L'espace typographique est féminin.

un canard¹
un ornithorynque¹⁸
un ours³²

1. bestiole qui fait coin
18. bestiole qui fait rire
32. bestiole qui fait mal

FIGURE 8.6 – Notes de bas de page (méthode avancée)

Les références internes

L^AT_EX vous permet d'écrire des références internes de façon simple. Pour ce faire, trois commandes sont à connaître. La commande `\label{nom_choisi}` sert à marquer un endroit, et les commandes `\ref{nom_choisi}` et `\pageref{nom_choisi}` vous permettent d'appeler le numéro de page ou la référence de l'élément marqué dans une autre zone de votre document (figure 8.7).

Démonstration :

```
\documentclass{report}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}

\begin{document}

\part{Partie}
\chapter{Chapitre}
\section{Section}
\subsection{Une sous-section}
\label{patate}
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Donec nec condimentum libero. Phasellus condimentum porttitor congue.
Morbi eget quam sed justo egestas lobortis. Aenean et erat metus.
\chapter{Chapitre}
\section{Section}
\subsection{Une sous-section}
Ici nous parlerons de la sous-section \ref{patate} vue à la page \pageref{patate}.
\end{document}
```

▷ Copier ce code
Code web : 334186

2.1 Section

2.1.1 Une sous-section

Ici nous parlerons de la sous section 1.1.1 vue à la page 2 .

FIGURE 8.7 – Référence interne

En résumé

- Les environnements `quote` et `quotation` permettent de citer du texte.
- On peut insérer du code source avec la commande `\verb` (dans un paragraphe), l'environnement `verbatim` (pour les codes plus longs) ou `lstlistings`⁶ (plus d'options).
- Il est possible de créer des boîtes de texte encadrées en combinant la commande `\fbox` et l'environnement `minipage`.
- Les notes de bas de page sont généralement insérées via `\footnote` mais peuvent être agencées plus finement grâce à `\footnotemark` et `\footnotetext`.
- Des marqueurs invisibles (`\label`) peuvent être disposés un peu partout dans les publications et être utilisés afin de créer des références internes⁷ avec `\ref` (indique le numéro de section où se trouve le marqueur) et `\pageref` (indique le numéro de page où se trouve le marqueur).

6. `lstlistings` offre de nombreuses options permettant notamment de colorer le code et de numérotter les lignes.

7. Nous apprendrons un peu plus loin à insérer des marqueurs sur des images ou des tableaux.

Chapitre 9

TP 1 – Mise en page de la Déclaration universelle des droits de l’homme

Difficulté : 

Il y a de ça huit chapitres, $\text{\LaTeX}$ envahissait votre vie. Vous avez appréhendé tous les concepts de base du langage, de la mise en page jusqu’à la hiérarchisation du contenu en passant par la personnalisation du texte et la production de citations élégantes. Toutes ces notions vous ont été enseignées de façon indépendante, mais il ne faut pas oublier la finalité de l’apprentissage de $\text{\LaTeX}$: la création de documents réels.

Ce premier TP est l’occasion pour vous de tester votre savoir-faire et de vous entraîner. L’intégralité du contenu des chapitres vus jusqu’ici sera mise à profit et il y a fort à parier que vous aurez besoin d’aller les consulter pour répondre aux problématiques de l’exercice.



Comme vous avez dû vous en douter, le titre du chapitre ne laissant aucun suspense, je vous annonce que nous allons mettre en page une publication sur la Déclaration universelle des droits de l'homme et du citoyen. Texte fondateur de notre civilisation moderne, il vous permettra de conjuguer apprentissage et culture.

Les consignes

Dans ce TP, vous allez travailler avec trois textes différents, en utiliser certains morceaux choisis et les mettre en page de façon à en faire un petit livre de 17 pages. Le nombre peut faire peur, mais ne vous inquiétez pas, ce travail devrait vous prendre moins d'une heure et demie si vous avez suivi les précédents chapitres avec attention.

Le document produit à l'issue de ce TP est téléchargeable au format PDF via le code web ci-dessous. Il est sobre, mais élégant, de quoi donner le sourire à vos lecteurs.

▷ Résultat du TP (PDF)
Code web : 539621

Le but ici est de vous faire utiliser des packages ainsi que de vous faire revoir les chapitres de la deuxième partie. Suite à l'exercice pratique, de multiples pistes vous seront données pour vous permettre d'aller plus loin dans votre apprentissage de L^AT_EX.

Autre chose à noter : **ce TP clôt la deuxième partie du cours (et vous avez survécu jusqu'ici).**

La partie III, qui suit ce TP, développe les commandes avancées de L^AT_EX telles que celles qui permettent de manipuler les images, les tableaux, les notations mathématiques et les tables. Bien qu'accessibles, ces notions nécessitent d'être à l'aise avec les notions de commandes, d'environnements et de packages. L'importance de travailler ce TP ne s'en trouve que renforcée.

Faites chauffer votre clavier : les consignes vous attendent dans le paragraphe ci-dessous. Elles sont longues, mais comportent une grande quantité de texte à prendre depuis les articles originaux de Wikipédia.



Les mots en gras ne le sont pas pour faire joli, mais pour vous aider !

Les contraintes de l'exercice

Contenu hiérarchisé de notre document

Votre travail consistera ici à créer un **livre** en **français** intitulé « **Les droits de l'homme** ».

Ce livre utilisera des extraits de textes issus de quatre sources différentes :

– l'article « Droits de l'Homme » de Wikipédia ;

- l'article « Déclaration des droits de l'Homme et du Citoyen de 1789 » de Wikipédia ;
- l'article « Révolution française » de Wikipédia ;
- le texte descriptif de la licence Creative Commons 3.0.

Vous pouvez récupérer ces articles pour les copier-coller facilement à l'aide du code web ci-après :

▷ Copier les articles
Code web : 285291

Sur la **page de garde** seront écrits :

- le titre du livre ;
- vos **nom** et prénom ;
- la date du jour.

Un **chapitre** nommé « Introduction » sera composé en préambule **gros et en italique**.

Il contiendra le texte suivant :

Les droits de l'homme sont un concept selon lequel tout être humain possède des droits universels, inaliénables, quel que soit le droit positif en vigueur ou les autres facteurs locaux tels que l'ethnie, la nationalité, l'orientation sexuelle ou la religion.

Selon cette philosophie, combattue ou éclipsée aux 19^e siècle, 20^e siècle et 21^e siècle par d'autres doctrines, l'homme, en tant que tel, et indépendamment de sa condition sociale, a des droits « inhérents à sa personne, inaliénables et sacrés », et donc opposables en toutes circonstances à la société et au pouvoir. Ainsi le concept de droits de l'homme est-il par définition universaliste et égalitaire, incompatible avec les systèmes et les régimes fondés sur la supériorité ou la « vocation historique » d'une caste, d'une race, d'un peuple, d'une classe ou d'un quelconque groupe social ; incompatible tout autant avec l'idée que la construction d'une société meilleure justifie l'élimination ou l'oppression de ceux qui sont censés faire obstacle à cette édification. Les droits de l'homme, types de prérogatives dont sont titulaires les individus, sont généralement reconnus dans les pays occidentaux par la loi, par des normes de valeur constitutionnelle ou par des conventions internationales, afin que leur respect soit assuré, si besoin est même contre l'État. L'existence, la validité et le contenu des droits de l'homme sont un sujet permanent de débat en philosophie et en sciences politiques.

Le **corps du livre** sera composé d'une partie intitulée « Déclaration des droits de l'homme et du citoyen de 1789 » contenant **trois chapitres**. Le **premier chapitre**, intitulé « Histoire », contiendra le texte ci-dessous :

La Déclaration des Droits de l'Homme a été écrite dans un temple protestant. L'assemblée réunie à Versailles par la convocation des États généraux pour trouver une solution fiscale au déficit de l'État, se déclare Assemblée nationale en réunissant les trois ordres, dont elle décide l'abolition, puis s'institue Assemblée nationale constituante, et décide de rédiger une déclaration des principes fondamentaux à partir desquels sera établie

une nouvelle Constitution. Elle se réunit pour cela, après avoir pris les décrets des 4 et 11 août 1789 sur la suppression des droits féodaux, qu'elle reprendra dans l'article premier de la Déclaration.

La Déclaration des Droits de l'Homme et du Citoyen a été débattue par l'Assemblée nationale française sous la présidence du marquis de Mirabeau à partir d'un des trois projets proposés : celui de 24 articles, rédigé par le VI^e bureau dirigé par Jérôme Champion de Cicé. L'attribution du texte primitif à La Fayette inspiré par la Déclaration d'indépendance des États-Unis est donc erronée. L'abbé Grégoire propose que la Déclaration des Droits de l'Homme et du Citoyen soit accompagnée de celle des devoirs.

La discussion débute le 9 juillet et débouche sur un vote le 26 août 1789, sous l'influence des leaders du tiers-état et de la noblesse. Le texte est peu modifié, mais est enrichi d'un préambule. Il est ratifié en partie le soir du 5 octobre 1789 par Louis XVI à Versailles, sur l'exigence de l'Assemblée, qui utilisa la pression d'une foule vindicative venue de Paris, initialement pour d'autres revendications. Entièrement promulguée par le Roi à Paris, le 3 novembre 1789, la Déclaration des Droits est la dernière ordonnance royale. Elle servira de Préambule à la première Constitution de la Révolution française.

Le **deuxième chapitre**, nommé « Le texte », est plus complexe et hiérarchisé. Sa première **section**, intitulée « Introduction », contient l'extrait ci-dessous :

Les représentants du peuple français, constitués en Assemblée nationale, considérant que l'ignorance, l'oubli ou le mépris des droits de l'homme sont les seules causes des malheurs publics et de la corruption des gouvernements, ont résolu d'exposer, dans une déclaration solennelle, les droits naturels, inaliénables et sacrés de l'homme, afin que cette déclaration, constamment présente à tous les membres du corps social, leur rappelle sans cesse leurs droits et leurs devoirs ; afin que les actes du pouvoir législatif et ceux du pouvoir exécutif, pouvant être à chaque instant comparés avec le but de toute institution politique, en soient plus respectés ; afin que les réclamations des citoyens, fondées désormais sur des principes simples et incontestables, tournent toujours au maintien de la Constitution et au bonheur de tous.

Sa seconde **section**, intitulée « Les articles », contient 17 **paragraphes**. Le premier paragraphe a pour titre « Article premier », les suivants s'appellent « Article 2 », « Article 3 »... « Article 17 ». Ci-dessous le texte respectif de chaque article. Prenez le temps de le lire, ne serait-ce que par intérêt culturel.

Article premier - Les hommes naissent et demeurent libres et égaux en droits. Les distinctions sociales ne peuvent être fondées que sur l'utilité commune.

Article 2 - Le but de toute association politique est la conservation des droits naturels et imprescriptibles de l'homme. Ces droits sont la liberté, la propriété, la sûreté et la résistance à l'oppression.

Article 3 - Le principe de toute souveraineté réside essentiellement dans la Nation. Nul corps, nul individu ne peut exercer d'autorité qui n'en émane expressément.

Article 4 - La liberté consiste à pouvoir faire tout ce qui ne nuit pas à autrui : ainsi, l'exercice des droits naturels de chaque homme n'a de bornes que celles qui assurent aux autres membres de la société la jouissance de ces mêmes droits. Ces bornes ne peuvent être déterminées que par la loi.

Article 5 - La loi n'a le droit de défendre que les actions nuisibles à la société. Tout ce qui n'est pas défendu par la loi ne peut être empêché, et nul ne peut être contraint à faire ce qu'elle n'ordonne pas.

Article 6 - La loi est l'expression de la volonté générale. Tous les citoyens ont droit de concourir personnellement ou par leurs représentants à sa formation. Elle doit être la même pour tous, soit qu'elle protège, soit qu'elle punisse. Tous les citoyens, étant égaux à ses yeux, sont également admissibles à toutes dignités, places et emplois publics, selon leur capacité et sans autre distinction que celle de leurs vertus et de leurs talents.

Article 7 - Nul homme ne peut être accusé, arrêté ou détenu que dans les cas déterminés par la loi et selon les formes qu'elle a prescrites. Ceux qui sollicitent, expédient, exécutent ou font exécuter des ordres arbitraires doivent être punis ; mais tout citoyen appelé ou saisi en vertu de la loi doit obéir à l'instant ; il se rend coupable par la résistance.

Article 8 - La loi ne doit établir que des peines strictement et évidemment nécessaires, et nul ne peut être puni qu'en vertu d'une loi établie et promulguée antérieurement au délit, et légalement appliquée.

Article 9 - Tout homme étant présumé innocent jusqu'à ce qu'il ait été déclaré coupable, s'il est jugé indispensable de l'arrêter, toute rigueur qui ne serait pas nécessaire pour s'assurer de sa personne doit être sévèrement réprimée par la loi.

Article 10 - Nul ne doit être inquiété pour ses opinions, mêmes religieuses, pourvu que leur manifestation ne trouble pas l'ordre public établi par la loi.

Article 11 - La libre communication des pensées et des opinions est un des droits les plus précieux de l'homme ; tout citoyen peut donc parler, écrire, imprimer librement, sauf à répondre de l'abus de cette liberté dans les cas déterminés par la loi.

Article 12 - La garantie des droits de l'homme et du citoyen nécessite une force publique ; cette force est donc instituée pour l'avantage de tous, et non pour l'utilité particulière de ceux à qui elle est confiée.

Article 13 - Pour l'entretien de la force publique, et pour les dépenses d'administration, une contribution commune est indispensable ; elle doit être également répartie entre les citoyens, en raison de leurs facultés.

Article 14 - Les citoyens ont le droit de constater, par eux-mêmes ou par leurs représentants, la nécessité de la contribution publique, de la consentir librement, d'en suivre l'emploi, et d'en déterminer la quotité, l'assiette, le recouvrement et la durée.

Article 15 - La société a le droit de demander compte à tout agent public

de son administration.

Article 16 - Toute société dans laquelle la garantie des droits n'est pas assurée ni la séparation des pouvoirs déterminée, n'a point de Constitution.

Article 17 - La propriété étant un droit inviolable et sacré, nul ne peut en être privé, si ce n'est lorsque la nécessité publique, légalement constatée, l'exige évidemment, et sous la condition d'une juste et préalable indemnité.

Le **troisième chapitre**, intitulé « Sources », contient le texte suivant.

La question des sources de la Déclaration française a suscité une controverse empreinte de nationalisme au sein de l'historiographie. Dans une brochure de 1895, l'historien allemand Georg Jellinek présentait l'œuvre française comme une simple héritière des Déclarations anglo-saxonnes (Pétition des droits, Déclaration des droits), elles-mêmes inspirées du Protestantisme luthérien. Traduite en français en 1902, dans un contexte de montée des tensions entre la France et l'Allemagne, elle donnera lieu à une réplique aussi peu nuancée, portée par Emile Boutmy : la Déclaration des Droits de l'Homme et du Citoyen n'aurait de source que dans la tradition philosophique et humaniste des Lumières.

Le Préambule, ajouté au projet, a été rédigé sous l'influence de Mirabeau et de Jean-Joseph Mounier, député du Tiers qui avait fait adopter le serment du Jeu de Paume, tous deux monarchiens, c'est-à-dire partisans d'une Monarchie constitutionnelle à l'anglaise. L'invocation à « l'Être suprême » a été rajoutée au cours des séances pour tenir compte des convictions chrétiennes de presque tous les citoyens.

Le texte de l'Article premier, « Tous les Hommes naissent et demeurent libres et égaux en droits », synthétise la Loi du 4 août 1789 abolissant la société d'ordres hiérarchisés.

L'article 16, associant constitution et organisation de la séparation des pouvoirs, est un principe antérieurement admis avec la séparation des ordres spirituel, politique et économique. Mais les trois pouvoirs politiques auxquels renvoie implicitement cet article, à savoir le législatif, l'exécutif et le judiciaire, est la conception proposée par Montesquieu depuis 1748 dans *De l'Esprit des Loïs*.

L'article 3, qui attribue la souveraineté à la Nation, s'inspire des thèmes des remontrances des Parlements, portées par les nombreux membres du club des Amis de la Constitution, plus connu sous le nom de Club des Jacobins, mais aussi du célèbre pamphlet de l'abbé Sieyès, qui pose l'équation : peuple = Tiers-État, c'est-à-dire que les députés du Tiers-État sont les représentants légitimes de l'ensemble du peuple.

L'article 6, directement inspiré du philosophe Rousseau, a été proposé par Talleyrand. Lu à la tribune du comité de constitution le 12 septembre 1789, ce qui deviendra l'article 6 de la Déclaration des droits prenait la forme suivante : « La loi étant l'expression de la volonté générale, tous les citoyens ont droit de concourir personnellement ou par représentation à sa formation ; elle doit être la même pour tous ».

Les autres articles reprennent des principes généraux du droit ou de la procédure qui sont déjà établis, comme la positivité du droit, le caractère contradictoire des procédures, la non-rétroactivité des lois, etc.

Son idéal est l'individualisme libéral. C'est une œuvre de circonstance, une proclamation générale, un texte tourné vers le passé avec pour objectif d'en finir avec l'Ancien Régime ; mais également un texte tourné vers l'avenir en promouvant la philosophie des Lumières et son idéal rationaliste.

L'annexe A, « La révolution française » nous raconte une belle histoire via ce texte :

La Révolution française est la période de l'histoire de France comprise entre la convocation des États généraux en 1789 et le coup d'État du 18 brumaire (9-10 novembre 1799) de Napoléon Bonaparte. C'est un moment fondamental de l'histoire de France, marquant la fin de l'Ancien Régime, et le passage à une monarchie constitutionnelle puis à la Première République. Elle a mis fin à la royauté, à la société d'ordres et aux privilèges. Justifiée par la Déclaration des Droits de l'Homme et du Citoyen, qui proclamait l'égalité théorique des citoyens devant la loi, les libertés fondamentales et la souveraineté de la Nation, apte à se gouverner au travers des représentants élus, cette période causa la mort de plusieurs milliers de personnes et la terreur pour la majorité.

Le chapitre épilogue sera nommé « Sources et licences », il contiendra **deux sections non numérotées**. La première section (« Sources ») contiendra une **liste de description**. La seconde (« Licence Creative Commons 3.0 ») comportera deux paragraphes. **Les mentions « Paternité » et « Partage des conditions initiales à l'identique »** devront être signalées comme importantes.

La mise en forme finale de la page devra être similaire à celle de la figure 9.1, avec bien sûr les adresses internet réelles et cliquables.

Consignes additionnelles et pistes

- Le pack de police utilisé sera **lmodern**.
- Les noms propres seront écrits en petites capitales (convention française).
- Les en-têtes et pieds de pages seront de type **headings** (réglage par défaut pour la classe **book**).
- La marge de droite sera de 3 cm.
- La marge de gauche sera de 6 cm.
- La marge du bas sera de 5 cm.
- La marge du haut sera de 5 cm.
- Les adresses internet ne devront pas tenir sur plus d'une ligne, mais vous avez le droit d'utiliser un réducteur d'URL comme **bit.ly** qui créera une adresse courte à partir d'une adresse initiale.

Les consignes s'arrêtent (enfin) ici !

Sources et licences

Sources

Droits de l'homme (Wikipedia) : <http://adresse.com>

Déclaration de 1789 (Wikipedia) : <http://adresse.com>

Déclaration de 1793 (Wikipedia) : <http://adresse.com>

Déclaration de 1795 (Wikipedia) : <http://adresse.com>

La révolution française (Wikipedia) : <http://adresse.com>

Licence Creative Commons 3.0

Vous êtes libres :

- de reproduire, distribuer et communiquer cette création au public ;
- de modifier cette création.

Selon les conditions suivantes :

- *Paternité* : vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'œuvre).
- *Partage des conditions initiales à l'identique* : si vous transformez ou modifiez cette œuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

FIGURE 9.1 – Mise en forme attendue

Afin de travailler dans de meilleures conditions, je vous conseille de compiler votre code très fréquemment pour régler les problèmes un par un. Quelques petits pièges se trouvent enfouis dans les textes de ce TP, à vous de les déjouer.

Tout est faisable avec vos connaissances actuelles, il suffit que vous preniez votre temps.

Au fait ! Ce tableau de caractères spéciaux devrait vous être utile...

ó	\'{o}	ö	\">{o}	ô	\~{o}
\	\textbackslash{}	õ	\~{o}	ō	\={o}
š	\d s	ó	\.{o}	ö	\u{o}
õ	\H{o}	ôo	\t{oo}	œ	\oe
q	\c{o}	o	\d{o}	š	\r s
o	\b{o}	Å	\AA	â	\aa
ß	\ss	ı	\i	ı	\j
š	\H s	ø	\o	š	\t s
š	\v s	Ø	\O	¶	\P
§	\S	æ	\ae	Æ	\AE
‡	\dag	‡	\ddag	©	\copyright
£	\pounds	{	\{	}	\}
%	\%	\$	\\$	&	\&
#	\#	—	_		
^	\~{}	...	\dots		



Dans l'encodage choisi, les apostrophes penchées provoquent des erreurs. À vous de trouver une solution...

Que la force soit avec vous !

La correction

L'heure est venue de corriger tout votre travail, de regarder les points qui ont pu poser des problèmes. Notez que chaque étape fait l'objet d'au moins une compilation dans le but de pouvoir corriger les erreurs pas à pas.

Étape 1 : le type de document

Le type de document a une importance prépondérante pour nous, il définit un squelette de code source. La classe `book` n'a pas été choisie par hasard ; souvenez-vous, nous avons traité ensemble les problématiques de la création d'annexes et de préambules.

La première étape du travail était de mettre en forme ce squelette, qui devait ressembler à peu de choses près à celui-ci :

```
| \documentclass{book}
```

```
\begin{document}

  \frontmatter
  \mainmatter
  \appendix
  \backmatter

\end{document}
```

Étape 2 : la langue

La consigne disait d'écrire le texte en français et il était visible que nombre d'accents allaient apparaître. Les trois packages habituels de gestion de caractères étaient donc indispensables. Il fallait les insérer dans le préambule du document, comme vous l'avez appris dans le chapitre traitant des packages.

```
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}
```

Étape 3 : les autres packages

Nous venons d'appeler les packages nous permettant d'écrire en français, il est de bon ton de survoler les consignes pour comprendre quels packages nous seront utiles pour continuer l'exercice. Ici, trois autres packages étaient nécessaires : `url`, `geometry` et `lmodern`.

```
\usepackage{url} % Pour écrire des adresses cliquables.
\usepackage{lmodern} % Pour changer le pack de police.
\usepackage[top=5cm, bottom=5cm, left=6cm, right=3cm]{geometry} % Les marges.
```

Étape 4 : la page de garde

La création de page de garde a été abordée dans les chapitres de mise en page. Celle demandée dans l'exercice est simple, elle comporte les trois mentions classiques : titre, nom des auteurs (les noms sont en petites capitales) et date du jour. Les trois paramètres sont définis dans le préambule et la commande `\maketitle` est insérée après `\begin{document}` afin de créer la page de garde (figure 9.2).

```
\title{Les droits de l'homme}
\author{\textsc{Laleloulilo} - \textsc{Zozor}}
\date{\today} % Pour mettre la date du jour, tapez \today.

\begin{document}

\maketitle % Page de garde.
```

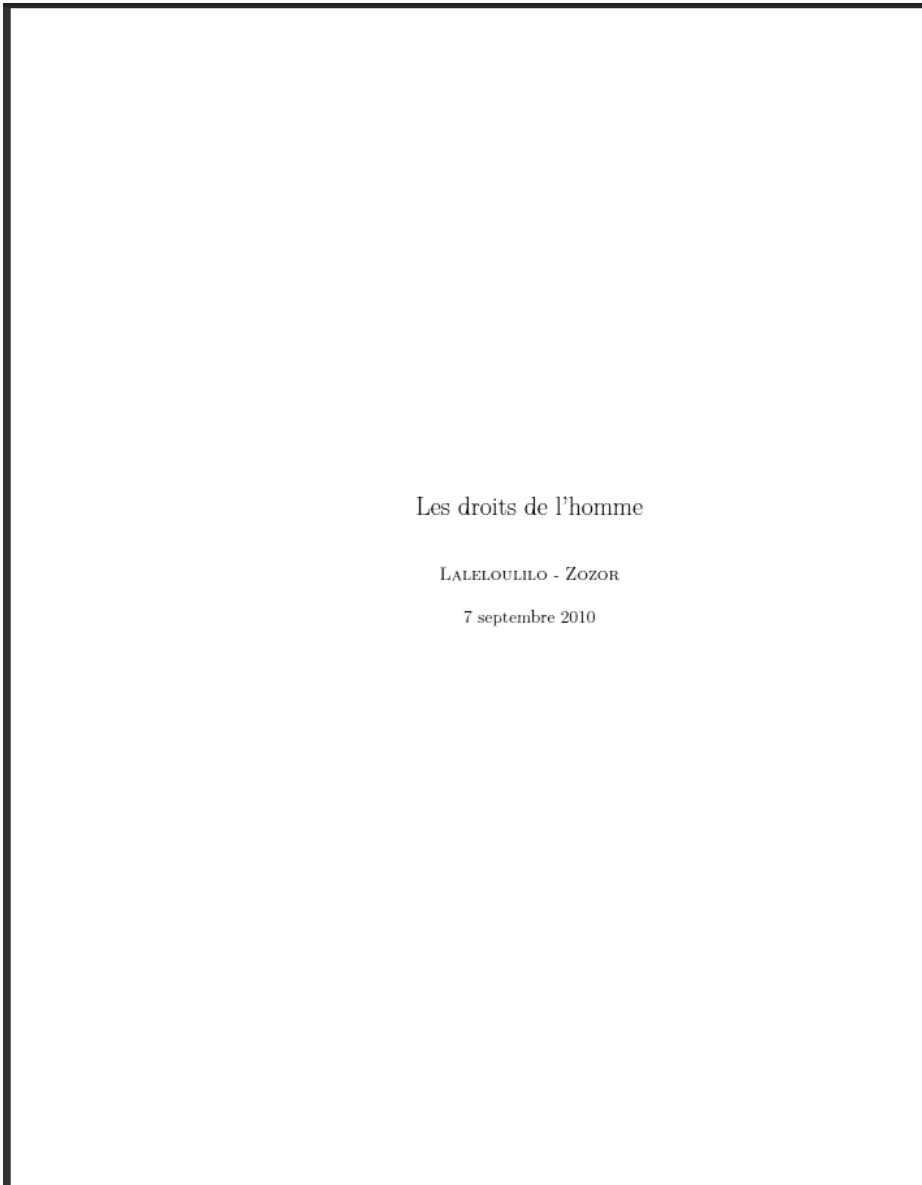


FIGURE 9.2 – Page de garde

Étape 5 : le squelette complet

Les packages sont listés, le travail préparatoire est presque terminé. Il nous reste les éléments de structure secondaires à lister (chapitres, sections...). Certains n'étant pas numérotés, nous n'oublions pas le symbole *.

```
\documentclass{book}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}
\usepackage{url} % Pour écrire des adresses cliquables.
\usepackage{lmodern} % Pour changer le pack de police.
\usepackage[top=5cm, bottom=5cm, left=6cm, right=3cm]{geometry} % Les marges.

\title{Les droits de l'homme}
\author{\textsc{Laleloulilo} - \textsc{Zozor}}
\date{\today} % Pour mettre la date du jour, tapez \today.

\begin{document}

\maketitle % Page de garde.

\frontmatter

\chapter{Introduction}

\mainmatter

\part{Déclaration des Droits de l'Homme et du Citoyen de 1789}

\chapter{Histoire}
\chapter{Le texte}

\section{Introduction}
\section{Les articles}

\paragraph{Article premier}
\paragraph{Article 2}
\paragraph{Article 3}
\paragraph{Article 4}
\paragraph{Article 5}
\paragraph{Article 6}
\paragraph{Article 7}
\paragraph{Article 8}
\paragraph{Article 9}
\paragraph{Article 10}
\paragraph{Article 11}
\paragraph{Article 12}
\paragraph{Article 13}
\paragraph{Article 14}
\paragraph{Article 15}
\paragraph{Article 16}
\paragraph{Article 17}

\chapter{Sources}
```

```

\appendix

\chapter{La révolution française}

\backmatter

\chapter{Sources et licences}

\section*{Sources}

\begin{description}
\item
\item
\item
\item
\item
\end{description}

\section*{Licence Creative Commons 3.0}

\paragraph{Vous êtes libres :}

\begin{itemize}
\item
\item
\end{itemize}

\paragraph{Selon les conditions suivantes :}

\begin{itemize}
\item \emph{Paternité} :
\item \emph{Partage des conditions initiales à l'identique} :
\end{itemize}

\end{document}

```

▷ Copier ce code
Code web : 945730

Si vous avez réussi à aller jusqu'ici, c'est déjà très bien. Félicitations, vous commencez à vous sentir à l'aise avec L^AT_EX et ses concepts fondamentaux !

Étape 6 : le contenu

Les choses sérieuses commencent. Ici se trouvent les plus grosses sources d'erreurs d'étourderie. Chaque chapitre en est rempli l'un après l'autre. Ne pas oublier d'insérer les commandes de mise en exposant, de création de noms en petites capitales, d'introduction de guillemets français ainsi que de remplacer par la commande \oe le caractère

spécial œ présent un peu partout dans les paragraphes.

Autre piège, certaines apostrophes sont penchées et font planter L^AT_EX (mais seulement si vous aviez tapé votre code en `latin1`). Il fallait que vous vous en rendiez compte et que vous les remplaciez par des apostrophes droites¹. Cette portion de code étant affreusement longue, vous pouvez la consulter grâce au code web suivant.

▷ Copier le contenu
Code web : 581319

Le chapitre d’introduction

Ce chapitre contenait des apostrophes penchées, des exposants ainsi que des guillemets français. De plus, il était souhaité qu’il soit gros et en italique (figures 9.3 et 9.4).

Les chapitres « Histoire », « Le texte », « Sources » et l’annexe « La Révolution française »

Rien de transcendant, les commandes utilisées ici l’ont déjà été dans le paragraphe précédent. La clé du succès était juste de prendre votre temps et de compiler souvent afin de mieux voir vos erreurs (figures 9.5 à 9.9).

Le chapitre épilogue « Sources et Licences »

Pour ce chapitre, il fallait dans un premier temps utiliser une liste de description ainsi que la commande `\url` accompagnée d’URL réduites via `bit.ly`². Dans un second temps, deux listes à puces toutes simples faisaient leur apparition. Dans l’une d’elles, la commande `\emph` sera manipulée pour signifier l’importance des termes « Paternité » et « Partage des conditions initiales à l’identique » (figure 9.10).

```
\chapter{Sources et Licences}

\section*{Sources}

\begin{description}
  \item[Droits de l’homme (Wikipédia) :] \url{http://bit.ly/9nYoSU}
  \item[Déclaration de 1789 (Wikipédia) :] \url{http://bit.ly/1DJqQg}
  \item[Déclaration de 1793 (Wikipédia) :] \url{http://bit.ly/SeRyG}
  \item[Déclaration de 1795 (Wikipédia) :] \url{http://bit.ly/b6sdRI}
  \item[La Révolution française (Wikipédia) :] \url{http://bit.ly/cUd3SF}
\end{description}

\section*{Licence Creative Commons 3.0}

\paragraph{Vous êtes libres :}
```

1. Ce genre de problème arrive de façon extrêmement fréquente au quotidien.

2. C’est un célèbre réducteur d’URL.

Introduction

Les droits de l'homme sont un concept selon lequel tout être humain possède des droits universels, inaliénables, quel que soit le droit positif en vigueur ou les autres facteurs locaux tels que l'ethnie, la nationalité, l'orientation sexuelle ou la religion.

Selon cette philosophie, combattue ou éclipsée aux 19^e siècle, 20^e siècle et 21^e siècle par d'autres doctrines, l'homme, en tant que tel, et indépendamment de sa condition sociale, a des droits « inhérents à sa personne, inaliénables et sacrés », et donc opposables en toutes circonstances à la société et au pouvoir. Ainsi le concept de droits de l'homme est-il par définition universaliste et égalitaire, incompatible avec les systèmes et les régimes fondés sur la supériorité ou la « vocation historique » d'une caste, d'une race, d'un peuple, d'une classe ou d'un quelconque groupe social; incompatible tout autant avec l'idée que la construction d'une société meilleure justifie l'élimination ou l'oppression de ceux qui sont censés faire obstacle à cette édification.

Les droits de l'homme, types de prérogatives dont sont titulaires les individus, sont généralement reconnus dans les pays occidentaux par la loi, par des normes de valeur

FIGURE 9.3 – Chapitre d'introduction

ii

INTRODUCTION

*constitutionnelle ou par des conventions internationales,
afin que leur respect soit assuré, si besoin est même contre
l'État. L'existence, la validité et le contenu des droits de
l'homme sont un sujet permanent de débat en philosophie
et en sciences politiques.*

FIGURE 9.4 – Chapitre d'introduction (la suite)

4 CHAPITRE 1. HISTOIRE	
Chapitre 1 Histoire La Déclaration des Droits de l'Homme a été écrite dans un temple protestant. L'assemblée réunie à Versailles par la convention des États généraux pour trouver une solution finale au déficit de l'État, se déclare Assemblée nationale constituante, et décide de rédiger une déclaration des principes fondamentaux, à partir de laquelle sera établie une nouvelle Constitution. Elle se réunit pour cela, après avoir pris les divers des 4 et 11 août 1789 sur la suppression des droits féodaux, qu'elle reprendra dans l'article premier de la Déclaration. La Déclaration des Droits de l'Homme et du Citoyen a été débattue par l'Assemblée nationale française sous la présidence du marquis de Mirabeau à partir d'un des trois projets proposés : celui de 21 articles, rédigé par le VP l'homme dirigé par Jérôme Chabotier et Cécil. L'attribution du texte principal à La Fayette inspiré par la Déclaration d'indépendance des États-Unis ont donc été erronés. L'abbé Grégoire propose que la Déclaration des droits de l'Homme et du Citoyen soit accompagnée de celle des devoirs. La discussion débute le 9 juillet et débouche sur un vote le 26 août 1789, sous l'auspice des bonnets du tiers-état et de la noblesse. Le texte est peu modifié, mais est adopté d'unanimité. Il est adopté en partie le soir du 5 octobre 1789 par Louis XVI à Versailles, sur l'insistance de l'Assemblée, qui utilise la pression d'une foule vindicative venue de Paris, initialement pour d'autres revendications. Entièrement promulguée par le Roi à Paris, le 3 novembre 1789, la Déclaration des Droits est la dernière ordonnance royale. Elle servira de Préambule à la première Constitution de la Révolution française. 3	

FIGURE 9.5 – Chapitre Histoire

6 CHAPITRE 2. LE TEXTE	
Chapitre 2 Le texte 2.1 Introduction Les représentants du peuple français, constitués en Assemblée nationale, considérant que l'ignominie, l'oubli ou le mépris des droits de l'homme sont les seules causes des maux publics et de la corruption des gouvernements, ont résolu d'écrire, dans une déclaration solennelle, les droits naturels, inaliénables et sacrés de l'homme, afin que cette déclaration, constamment présente à tous les membres du corps social, leur rappelle sans cesse leurs droits et leurs devoirs ; afin que les actes du pouvoir législatif et ceux du pouvoir exécutif, pouvant être à chaque instant comparés avec le but de toute institution politique, en soient plus respectés ; afin que les réclamations des citoyens, fondées désormais sur des principes simples et incontestables, tournent toujours au maintien de la Constitution et au bonheur de tous. 2.2 Les articles Article premier Les hommes naissent et demeurent libres et égaux en droits. Les distinctions sociales ne peuvent être fondées que sur l'utilité commune. Article 2 Les hommes naissent et demeurent libres et égaux en droits. Les distinctions sociales ne peuvent être fondées que sur l'utilité commune. Article 3 Le principe de toute souveraineté réside essentiellement dans la Nation. Nul corps, nul individu ne peut exercer d'autorité qui n'en émane expressément. 5	Article 4 La liberté consiste à pouvoir faire tout ce qui ne nuit pas à autrui ; ainsi, l'exercice des droits naturels de chaque homme n'a de bornes que celles qui assurent aux autres membres de la société la jouissance de ces mêmes droits. Ces bornes ne peuvent être déterminées que par la loi. Article 5 La loi n'a le droit de défendre que les actions nuisibles à la société. Tout ce qui n'est pas défendu par la loi ne peut être empêché, et nul ne peut être contraint à faire ce qu'elle n'ordonne pas. Article 6 La loi est l'expression de la volonté générale. Tous les citoyens ont droit de concourir personnellement ou par leurs représentants à sa formation. Elle doit être la même pour tous, soit qu'elle protège, soit qu'elle punisse. Tous les citoyens, étant égaux à ses yeux, sont également admissibles à toutes dignités, places et emplois publics, selon leur capacité et sans autre distinction que celle de leurs vertus et de leurs talents. Article 7 La loi est l'expression de la volonté générale. Tous les citoyens ont droit de concourir personnellement ou par leurs représentants à sa formation. Elle doit être la même pour tous, soit qu'elle protège, soit qu'elle punisse. Tous les citoyens, étant égaux à ses yeux, sont également admissibles à toutes dignités, places et emplois publics, selon leur capacité et sans autre distinction que celle de leurs vertus et de leurs talents. Article 8 La loi ne doit établir que des peines strictement et évidemment nécessaires, et nul ne peut être puni qu'en vertu d'une loi établie et promulguée antérieurement au délit, et légalement appliquée. Article 9 Tout homme étant présumé innocent jusqu'à ce qu'il ait été déclaré coupable, s'il est jugé indispensable de l'arrêter, toute rigueur qui ne serait pas nécessaire pour s'assurer de sa personne doit être sévèrement réprimée par la loi. Article 10 Nul ne doit être inquiété pour ses opinions, mêmes religieuses, pourvu que leur manifestation ne trouble pas l'ordre public établi par la loi. Article 11 La libre communication des pensées et des opinions est un des droits les plus précieux de l'homme : tout citoyen peut donc parler, écrire, imprimer librement, sauf à répondre de l'abus de cette liberté dans les cas déterminés par la loi.

FIGURE 9.6 – Chapitre Texte

CHAPITRE 9. TP 1 – MISE EN PAGE DE LA DÉCLARATION UNIVERSELLE
DES DROITS DE L’HOMME

2.2. LES ARTICLES Article 12 La garantie des droits de l'homme et du citoyen nécessite une force publique; cette force est donc instituée pour l'usage de tous, et non pour l'utilité particulière de ceux à qui elle est confiée. Article 13 Pour l'entretien de la force publique, et pour les dépenses d'administration, une contribution commune est indispensable; elle doit être également répartie entre les citoyens, en raison de leurs facultés. Article 14 Les citoyens ont le droit de constater, par eux-mêmes ou par leurs représentants, la nécessité de la contribution publique, de la consentir librement, d'en suivre l'emploi, et d'en déterminer la quotité, l'assiette, le recouvrement et la durée. Article 15 La société a le droit de demander compte à tout agent public de son administration. Article 16 Toute société dans laquelle la garantie des droits n'est pas assurée ni la séparation des pouvoirs déterminée, n'a point de Constitution. Article 17 La propriété étant un droit inviolable et sacré, nul ne peut en être privé; et ce n'est lorsque la nécessité publique, légalement constatée, l'exige évidemment, et sous la condition d'une juste et préalable indemnité.	8 CHAPITRE 2. LE TEXTE
--	--------------------------------------

FIGURE 9.7 – Chapitre Texte (bis)

Chapitre 3 Sources La question des sources de la Déclaration française a suscité une controverse importante de nationalisme au sein de l'historiographie. Dans une brochure de 1905, l'historien allemand Georg HEILNER présentait l'œuvre française comme une simple héritière des Déclarations anglo-américaines (Prétention des droits, Déclaration des droits), elle-même inspirée du Protestantisme luthérien. Traduite en français en 1902, dans un contexte de tensions entre France et Allemagne, elle donna lieu à une réplique aussi peu mesurée, portée par Emile BOUVIER : la Déclaration des droits de l'homme et du citoyen n'aurait de source que dans la tradition philosophique et humaniste des Lumières. Le Président, après un projet, a été rédigé sous l'influence de MIRABEAU, et de Jean-Joseph MOUNIER, député du Tiers qui avait fait adopter le serment de Jeu de Pume, tous deux monarchistes, c'est-à-dire partisans d'une Monarchie constitutionnelle à l'anglaise. L'association L'Ét - Être signifie n'a été ajoutée au texte des auteurs pour tenir compte des convictions chrétiennes de presque tous les citoyens. Le texte de l'article Un, « Tous les Hommes naissent et demeurent libres et égaux en droit », rappelle la Loi de l'unité L'Étre abolissant la société d'ordres hiérarchisés. L'article 16, concernant constitution et organisation de la séparation des pouvoirs, est un principe antérieurement admis avec la séparation des ordres spirituel, politique et économique. Mais les trois pouvoirs politiques auxquels renvoie implicitement cet article, à savoir le législatif, l'exécutif et le judiciaire, ont la conception proposée par Montesquieu depuis 1748 dans <i>De l'Esprit des Lois</i>. L'article 3, qui attribue la souveraineté à la Nation, s'inspire des thèmes des remontrances des Parlements, portées par les nombreux membres du club des Amis de la Constitution, plus ceux sous le nom de Club des Jacobins, mais aussi du célèbre pamphlet de l'abbé SIEYÈS, qui pose l'équation : peuple = Tiers-État, c'est-	10 CHAPITRE 3. SOURCES à-dire que les députés du Tiers-État sont les représentants légitimes de l'ensemble du peuple. L'article 6, directement inspiré du philosophe ROUSSEAU, a été proposé par TALLEYRAND. La 1^{re} version de comité de constitution le 12 septembre 1789, et qui deviendra l'article 6 de la déclaration des droits prévoit la formule suivante : « La loi étant l'expression de la volonté générale, tous les citoyens ont droit de concourir personnellement ou par représentation à sa formation : elle doit être la même pour tous ». Les autres articles reprennent des principes généraux du droit ou de la procédure qui sont déjà établis, comme la positivité du droit, le caractère contraignant des décisions, la non rétroactivité des lois, etc. Son idéal est l'individualisme libéral. C'est une œuvre de circonstance, une proclamation générale, un texte tourné vers le passé avec pour objectif d'en faire avec l'Ancien Régime; mais également un texte tourné vers l'avenir en promouvant la philosophie des lumières et son idéal rationaliste.
---	--

FIGURE 9.8 – Chapitre Sources

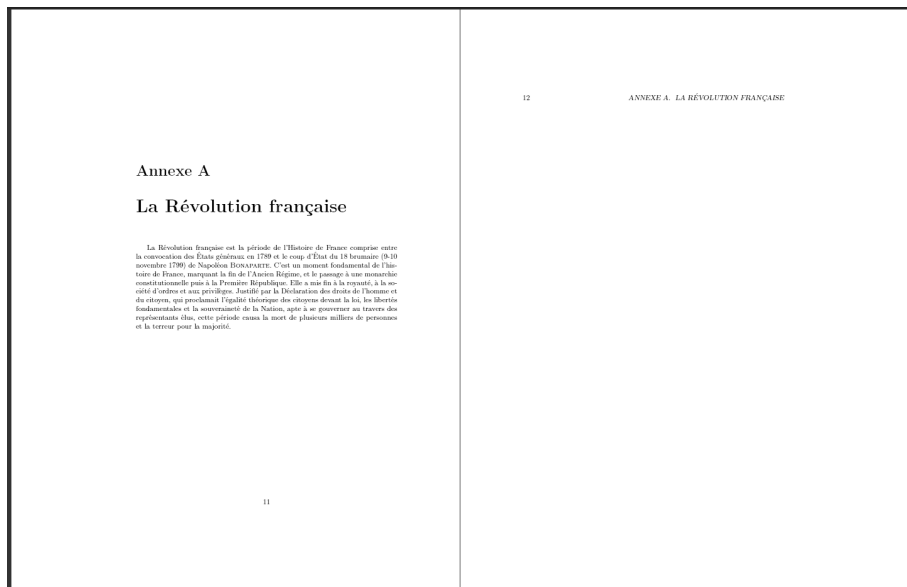


FIGURE 9.9 – Annexe A

```
\begin{itemize}
  \item de reproduire, distribuer et communiquer cette création au public ;
  \item de modifier cette création.
\end{itemize}

\paragraph{Selon les conditions suivantes :}

\begin{itemize}
  \item \emph{Paternité} : vous devez citer le nom de l'auteur original
  ↪ de la manière indiquée par l'auteur de l'\oe uvre ou le titulaire
  ↪ des droits qui vous confère cette autorisation (mais pas d'une manière qui
  ↪ suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de
  ↪ l'\oe uvre).
  \item \emph{Partage des conditions initiales à l'identique} :
  ↪ si vous transformez ou modifiez cette oeuvre pour en créer une nouvelle,
  ↪ vous devez la distribuer selon les termes du même contrat ou avec une
  ↪ licence similaire ou compatible.
\end{itemize}
```

Étape 7 : rassembler les morceaux

L'étape 7 est tout simplement celle au cours de laquelle nous allons regrouper tous les éléments du fichier (je l'ai appelé `tp.tex`, libre à vous de le nommer autrement). Une

Sources et licences

Sources

Droits de l'homme (Wikipédia) : <http://bit.ly/9nYoSU>

Déclaration de 1789 (Wikipédia) : <http://bit.ly/1DJqQg>

Déclaration de 1793 (Wikipédia) : <http://bit.ly/SeRyG>

Déclaration de 1795 (Wikipédia) : <http://bit.ly/b6sdRI>

La révolution française (Wikipédia) : <http://bit.ly/cUd3SF>

Licence Creative Commons 3.0

Vous êtes libres :

- de reproduire, distribuer et communiquer cette création au public ;
- de modifier cette création.

Selon les conditions suivantes :

- *Paternité* : vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'œuvre).
- *Partage des conditions initiales à l'identique* : si vous transformez ou modifiez cette œuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

FIGURE 9.10 – Sources

fois tous les morceaux rassemblés, le code source est complet. J'ai là aussi choisi de vous donner un code web plutôt que 20 pages de code illisibles.



La numérotation en première page d'un chapitre est différente de celle utilisée dans les pages qui suivent. C'est un réglage par défaut de $\text{\LaTeX}$: ce n'est ni une erreur, ni un bug.

▷ Code final complet
Code web : 797371

Conclusion

Vous voici arrivé à la fin de cette correction. Vous avez pu constater qu'une bonne organisation permettait d'avancer vite dans la création de documents avec $\text{\LaTeX}$. Seules 7 étapes plutôt simples ont été nécessaires pour venir à bout de ce TP.

Gardez ces étapes en tête, elles vous aideront à créer de longs rapports sans vous embrouiller³. Il est maintenant temps de vous présenter de **nouveaux outils**. Ces outils sont un peu plus complexes que ceux utilisés jusqu'à maintenant, mais vous seront utiles pour concevoir des documents à la mise en page plus fine.

Aller plus loin

Ce TP a été l'occasion de vous entraîner à manipuler des concepts simples. Dans cette sous-partie, nous allons voir quelques subtilités de $\text{\LaTeX}$ qui pourront vous servir lors de la conception de vos publications.

Commandes et espaces

Nous allons ici apprendre à mettre une espace après une commande $\text{\LaTeX}$. Petit test, compilez le code suivant⁴ (donnant la figure 9.11) :

```
\emph{canard} lapin\\
\LaTeX lapin\\
\LaTeX{} lapin\\
```

Constatez que dans la deuxième ligne, il n'y a pas d'espace entre les deux mots alors que vous en aviez inséré une dans le code. À la troisième ligne, vous remarquez que l'insertion de {} vient pallier ce problème. Nous allons donc formuler ici une règle simple qui vous permettra d'insérer des espaces après les commandes **se terminant par des lettres**.

3. Cela peut très vite arriver : lors de la rédaction du TP, je me suis aussi emmêlé les pinceaux.

4. La commande $\text{\LaTeX}$ sert à écrire le mot $\text{\LaTeX}$.

canard lapin
L^AT_EXlapin
L^AT_EX lapin

FIGURE 9.11 – Les espaces

Les commandes se terminant par des lettres doivent être suivies de {} afin de pouvoir insérer une espace à leur suite.

Règle toute simple, elle vous sauvera dans de nombreuses situations⁵. Nous apprendrons à gérer les espaces de manière beaucoup plus fine dans le second TP.

Marges et layout

Lors du TP, nous avons paramétré les marges de notre document de sorte qu'il ait des marges définies en haut, en bas, à gauche et à droite. Il est temps d'apprendre à faire les choses de manière plus fine. Lors de la création d'un layout⁶, L^AT_EX vous montre toute une liste de paramètres. Chaque paramètre d'un layout peut être modifié et nous allons apprendre ici à le faire.



Ces informations sont données à titre de précision : si vos besoins en termes de mise en page n'exigent pas un niveau de détail exceptionnel, vous pouvez très bien vous contenter de définir les marges en haut, en bas, à droite et à gauche comme nous l'avons fait jusqu'à maintenant.

Avant toute manipulation, remémorons-nous l'apparence d'un layout ainsi que le code source nécessaire à sa création (figure 9.12) :

```
\documentclass{article}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{layout}

\begin{document}

\layout

\end{document}
```

5. Je n'ai pas utilisé cette règle dans le TP, étant donné que nous ne l'avions pas encore vue.

6. Créé via la commande `\layout` que nous avons déjà vue dans le chapitre concernant les marges.

▷ Copier ce code
Code web : 801168

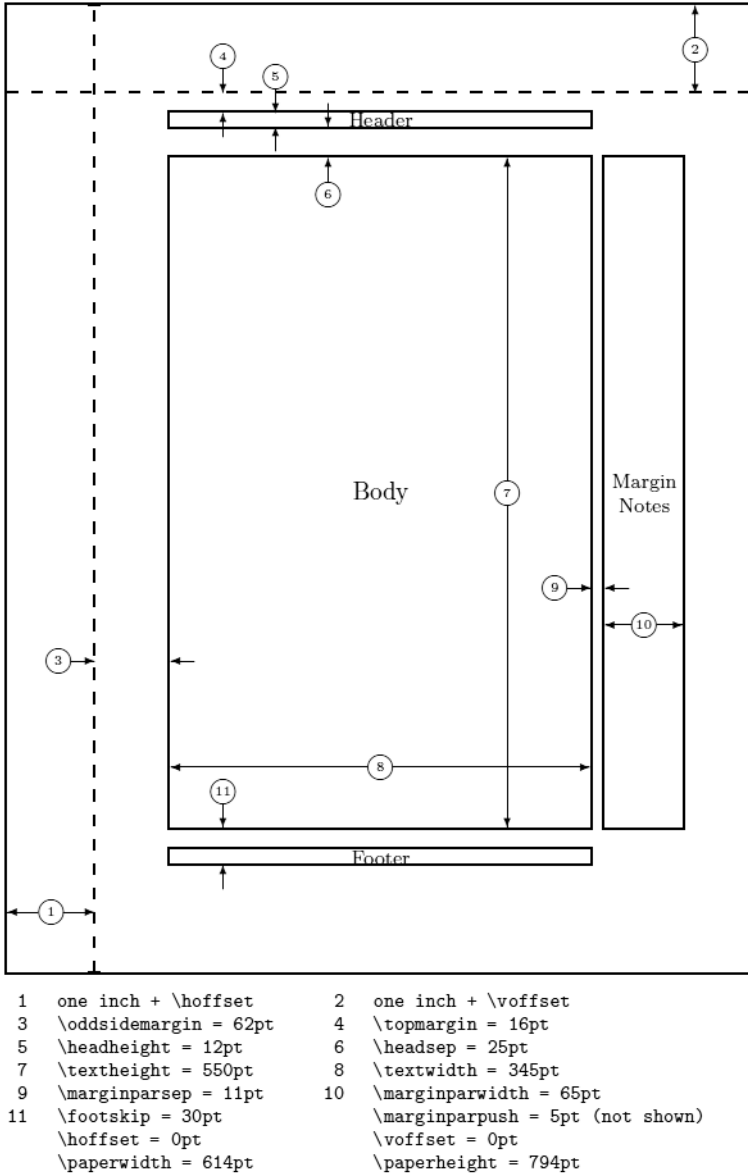


FIGURE 9.12 – Layout par défaut

Chaque longueur est associée à une commande et à un repère sur le layout, de sorte que vous puissiez facilement identifier les différentes longueurs et les associer à leur nom.

Je vous montre ici comment modifier une longueur du layout. Prenons la longueur

`\marginparwidth` repérée par le nombre 10 dans le layout (figure 9.12). Pour la modifier (figure 9.13), il suffit d'utiliser la commande suivante dans le préambule du document :

```
\setlength{nom de la longueur}{longueur dans l'unité qui vous plaît}  
\setlength{\marginparwidth}{2cm} % Les centimètres seront automatiquement  
→ convertis.
```

Ce mécanisme est applicable à toutes les longueurs figurant dans le layout.



La liste des unités de longueur vous sera fournie dans l'annexe « Mémento ».

Personnalisation de la commande `\emph`

Nous avons vu dans le chapitre traitant du texte et des polices que la commande `\emph` ne servait non pas à mettre du texte en italique, mais à mettre des termes en emphase (c'est-à-dire à signaler leur importance).

Ce TP est l'occasion pour vous d'apprendre à modifier la manière dont `\emph` met en valeur du texte. Nous souhaitons ici que la commande transforme le texte en texte type machine à écrire plutôt qu'en italique. Le résultat serait le suivant (capture d'écran du dernier paragraphe du TP sur la figure 9.14) :

```
\begin{itemize}  
  \item \emph{Paternité} : vous devez citer le nom de l'auteur original  
→ de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des  
→ droits qui vous confère cette autorisation (mais pas d'une manière qui  
→ suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de  
→ l'oeuvre).  
  \item \emph{Partage des conditions initiales à l'identique} : si  
→ vous transformez ou modifiez cette œuvre pour en créer une nouvelle, vous  
→ devez la distribuer selon les termes du même contrat ou avec une licence  
→ similaire ou compatible.  
\end{itemize}
```

Comment en sommes-nous arrivés là? Tout simplement en redéfinissant la commande `\emph` grâce à la commande suivante insérée dans le préambule du document :

```
\renewcommand{nom de la commande}{fonctions liées à cette commande}  
\renewcommand{\emph}{\texttt}
```

Ici la commande est `\emph`, elle est associée à la transformation « machine à écrire » (`\texttt`). N'ayant pas spécifié que le texte devait être en italique, il ne le sera pas.

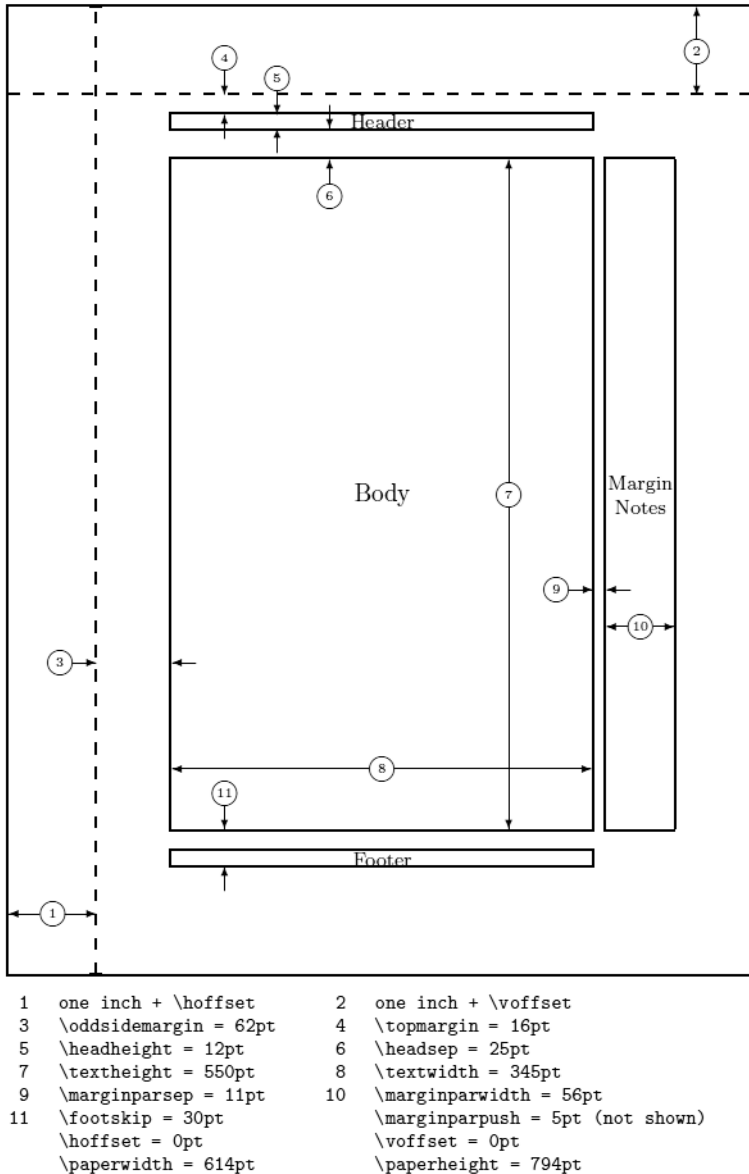


FIGURE 9.13 – Layout modifié

Selon les conditions suivantes :

- **Paternité** : vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'œuvre).
- **Partage des conditions initiales à l'identique** : si vous transformez ou modifiez cette oeuvre pour en créer une nouvelle, vous devez la distribuer selon les termes du même contrat ou avec une licence similaire ou compatible.

FIGURE 9.14 – Mise en valeur du texte

En revanche, des problèmes pourront se poser si elle est insérée dans un texte en italique (elle ne jouera plus son rôle). La commande `\renewcommand` a de nombreuses applications, nous en verrons d'autres par la suite.

En-têtes et pieds de pages personnalisés

Il y a de ça quelques chapitres, vous étiez jeune et innocent et utilisiez la commande `\pagestyle` avec l'une des trois options prédéfinies : `plain`, `empty` ou `headings`. Vous voici prêt à apprendre à créer des en-têtes et pieds de pages personnalisés. La commande `\pagestyle{fancy}` permet de faire cela (elle est fournie par le package `fancyhdr`). `\pagestyle{fancy}` découpe chaque en-tête et pied de page en trois morceaux, de sorte que nous devons personnaliser six commandes différentes pour arriver à créer une mise en page cohérente des en-têtes et pieds de pages.

- `\lhead{...}` gère le coin gauche de l'en-tête.
- `\chead{...}` gère le centre de l'en-tête.
- `\rhead{...}` gère le coin droit de l'en-tête.
- `\lfoot{...}` gère le coin gauche du pied de page.
- `\cfoot{...}` gère le centre du pied de page.
- `\rlfoot{...}` gère le coin droit du pied de page.

Diverses autres commandes sont utilisables :

- `\leftmark` : nom du chapitre ;
- `\rightmark` : nom de la section ;
- `\thepage` : numéro de page ;
- `\thechapter` : numéro du chapitre ;
- `\thesection` : numéro de la section courante ;
- `\renewcommand{headrulewidth}{épaisseur dans l'unité de votre choix}` : épaisseur de la ligne entre l'en-tête et le corps du texte ;
- `\renewcommand{footrulewidth}{épaisseur dans l'unité de votre choix}` : épaisseur de la ligne entre le pied de page et le corps du texte ;
- `\renewcommand{headsep}{longueur dans l'unité de votre choix}` : espacement entre l'en-tête et le corps du document.

Faites le test sur le document que nous venons de créer dans le TP en remplaçant les premières lignes par les suivantes.

```
\documentclass{book}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{url}
\usepackage{lmodern}
\usepackage{fancyhdr}
\usepackage[top=5cm, bottom=5cm, left=6cm, right=3cm]{geometry}

\title{Les droits de l'homme}
\author{\textsc{Laleloulilo} - \textsc{Zozor}}
\date{\today} % Pour mettre la date du jour, tapez \today.
\begin{document}

\pagestyle{fancy}

\renewcommand{\headheight}{13pt}
\renewcommand{\headrulewidth}{2pt}
\renewcommand{\footrulewidth}{2pt}
\renewcommand{\headsep}{15pt}
\lhead{\rightmark }
\chead{}
\rhead{\leftmark }
\lfoot{Section \thesection}
\cfoot{Chapitre \thechapter }
\rfoot{\thepage}

\maketitle % Page de garde.
```

▷

Copier ce code
Code web : 654093

Toutes les commandes ont été utilisées ici. Avouons-le, certaines pages ne sont pas des modèles de grâce. Qu'à cela ne tienne, vous pouvez changer le style des en-têtes et pieds de pages à la carte via la commande `\thispagestyle{}` qui peut se voir attribuer elle aussi les options `plain`, `empty` ou `headings`. Cette commande se place n'importe où dans le document, elle prendra effet sur la page où le texte qui l'entoure est mis en page.

En résumé

- Écrire un petit livre en $\text{\LaTeX}$ n'est pas difficile, mais demande beaucoup d'organisation. Il faut respecter la hiérarchie imposée par $\text{\LaTeX}$.

- La commande `\emph{}` indique que le texte doit être mis en valeur. Celui-ci peut être écrit en italique ou d'une façon différente selon le contexte.
- Les en-têtes et pieds de pages peuvent être paramétrés grâce à `\pagestyle{fancy}` et la foule de commandes annexes qui lui sont dédiées.
- Les commandes se terminant par des lettres doivent être suivies de `{}` si l'on veut insérer une espace à leur suite.

Troisième partie

Utilisation avancée de LaTeX

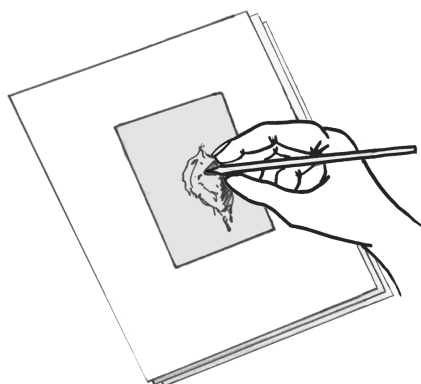
Chapitre 10

Les figures

Difficulté : 

Ce premier chapitre de la troisième partie de cet ouvrage introduit les manipulations de figures. Les concepts et packages présentés ici, bien qu'assez faciles à prendre en main dans des cas simples, peuvent se révéler extrêmement agaçants dans des situations plus exotiques. Les plus expérimentés connaissent la difficulté rencontrée lors de l'insertion de certaines figures dans des documents à la mise en page complexe.

Ne vous inquiétez pas, nous allons découvrir ensemble et sans empressement les notions liées aux figures. Un nouveau concept fera également son apparition dans ce livre : **le flottant**. Nous prendrons le temps de l'étudier.



Les différents formats d'images

L^AT_EX, avec ses packages, vous permet d'insérer des figures et des images de divers formats dans votre document et de les arranger selon vos souhaits. Une chose importante est à noter : **tous les formats d'images ne sont pas compatibles entre eux**.

Pourquoi? À cause de la compilation. Nous avons appris dans le troisième chapitre qu'un fichier `.tex` pouvait, selon vos besoins, être compilé en fichiers de différents formats (nous avons parlé des formats PS et PDF). Selon la méthode de compilation, il vous sera permis d'utiliser certains formats d'images.

Avant de donner des explications techniques, jetons un œil au schéma de la figure 10.1.

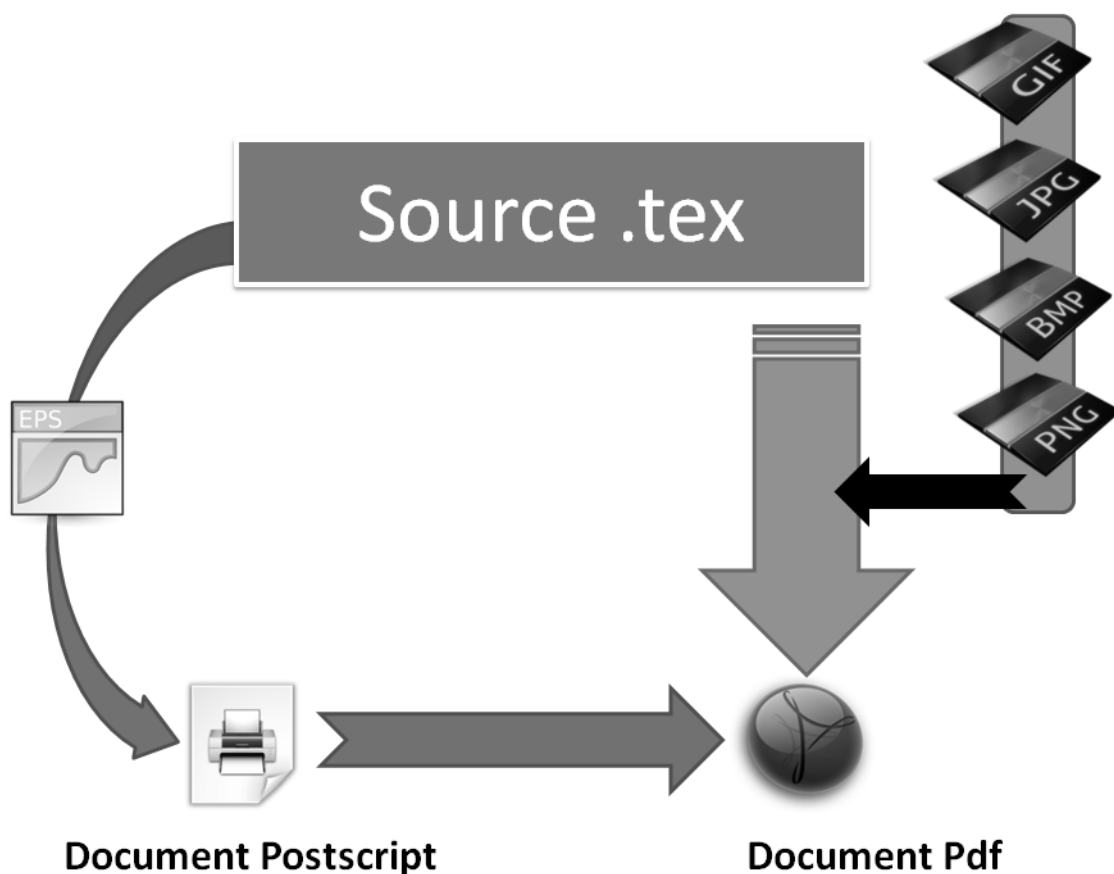


FIGURE 10.1 – La gestion des formats d'images

Nous nous intéresserons ici à l'insertion de plusieurs formats courants d'images et de

figures (EPS, PNG, GIF, JPEG, BMP). Nous constatons sur ce schéma qu'un fichier PDF s'obtient de deux manières bien distinctes : soit il est le produit d'une compilation directe du fichier `.tex`, soit il est le résultat de la conversion d'un fichier PostScript (ces possibilités de conversion et de compilation ont été abordées dans le chapitre 3 en détail, revenez-y si elles ne sont pas claires dans votre esprit).

Cette différence nous intéresse, parce que le schéma montre qu'une compilation directe en PDF ne permet pas l'insertion de figures d'extension `.eps`, alors qu'une compilation d'un fichier `.tex` en un fichier PostScript l'autorise.

Nous sommes ici face à une originalité technique embêtante : tous les formats d'images ne peuvent pas être utilisés simultanément dans un document. Conséquence logique, un choix entre l'utilisation de figures¹ (EPS et PS) ou d'images (BMP, GIF, PNG, JPEG) s'impose dès la conception d'un document.



Une image ou une figure peut se convertir facilement d'un format à un autre en l'ouvrant avec un logiciel de traitement d'images (Gimp, Photoshop...) pour l'enregistrer sous un autre format.

Votre choix doit se faire selon le type de fichier que vous manipulez. Des chercheurs auront plutôt tendance à travailler avec le format PostScript, alors que les autres préféreront probablement utiliser des images au format PNG. Quel que soit votre choix², les packages et les commandes utilisés par la suite restent les mêmes.

Résumons tout ceci !

- Si vous souhaitez insérer des figures en `.eps`, il est **obligatoire** de compiler en PostScript avant d'effectuer une conversion en PDF (vous pouvez également conserver votre fichier en PS). Il vous sera alors interdit d'utiliser des formats tels que le PNG, le BMP, le JPEG ou le GIF.
- Si vous souhaitez insérer des images (GIF, BMP, JPEG, PNG...), il est **obligatoire** de compiler en PDF **directement**. Il vous sera interdit d'utiliser les figures au format PS ou EPS.
- Une image peut facilement se convertir en figure et vice-versa.

Insertion d'images et de figures

L'épineux problème de la cohabitation entre les fichiers `.eps` et les autres formats d'images ayant été traité, nous pouvons aborder la pratique.

Nous allons apprendre à insérer une ou plusieurs images dans un document. Les opérations effectuées dans cette partie nécessiteront toutes l'utilisation du package `graphicx`. Chargez-le via la commande `\usepackage{graphicx}` à chaque fois que vous aurez à manipuler des images.

1. Pour des raisons de compréhension, j'appelle « figures » la famille des PostScript et « images » la famille des PNG. Même si le choix est assez litigieux, au moins il m'est plus facile de vous faire comprendre quels formats sont autorisés (ou ne le sont pas) dans une publication en cours.

2. En cas d'hésitation, je vous conseille plutôt les PNG.

Pour travailler une image, il nous faut tout d'abord... une image. Voici donc la Wonder Poulpy³, qui a accepté de se faire photographier en exclusivité afin que nous puissions manipuler sa photo et l'inclure dans nos documents (figure 10.2).

▷ Télécharger l'image
Code web : 307385



FIGURE 10.2 – Poulpy



Téléchargez cette image grâce au code web fourni ci-dessus. Pour ce faire, effectuez un clic droit sur l'image avant de l'enregistrer sous le nom de votre choix (évités les accents et les espaces) **dans le dossier où vous composez vos documents L^AT_EX**. Pour ma part, je l'ai appelée `poulpy.png`.

Insertion simple

Une image est insérée dans un document via la commande suivante :

```
| \includegraphics{chemin de l'image}
```

Le chemin de l'image est l'adresse de l'image sur votre ordinateur. Vous pouvez spécifier un chemin absolu ou un chemin relatif, au choix.



Le chemin absolu est le chemin exact à parcourir dans vos dossiers et répertoires permettant de retrouver votre image, il est du type : `C:\Cassoulet\Documents\fichierslatex\chapitre9\poulpy.png`.

Le chemin relatif, en revanche, est le chemin à parcourir depuis le fichier source pour retrouver l'image. Ainsi, si le fichier `.tex` se trouve dans le répertoire `fichierslatex` mentionné précédemment, le chemin relatif de l'image sera `chapitre9\poulpy.png`.

Comme nous avons enregistré l'image de Poulpy dans le dossier où se trouve notre fichier `.tex`, son chemin relatif se résume à `poulpy.png`. C'est celui-ci que nous utiliserons

3. Mascotte officielle de tous les jours et unique modèle de poulpe de compagnie.

dans l'exemple qui suit :

```
\documentclass[12pt]{report}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{graphicx}

\begin{document}
\includegraphics{poulpy.png}
\end{document}
```

▷ Copier ce code
Code web : 256203

Le résultat attendu s'affiche : l'image est à sa taille d'origine et positionnée en haut à gauche du document (figure 10.3).

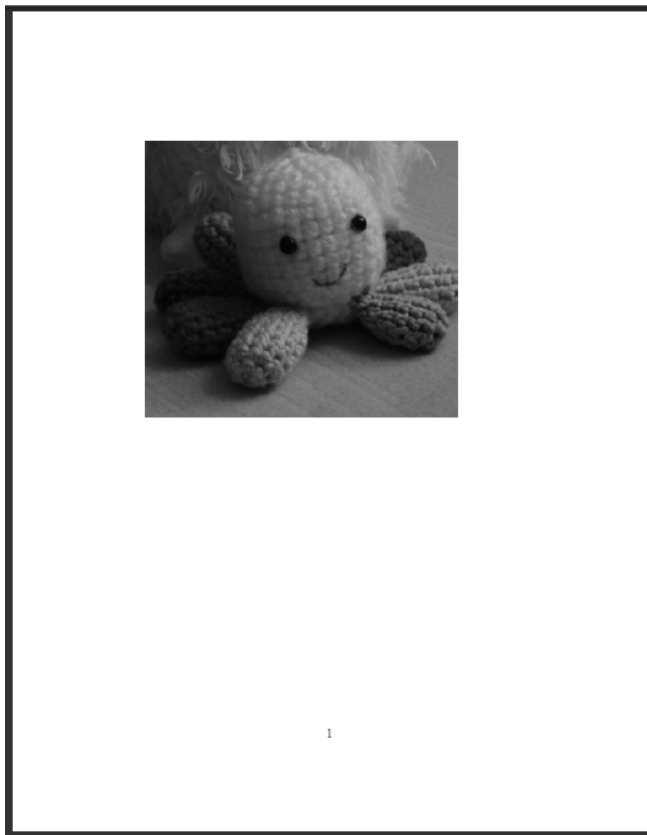


FIGURE 10.3 – Insertion de Poulpy

Taille d'une image

Vous avez plusieurs possibilités pour indiquer à L^AT_EX la taille de l'image que vous souhaitez insérer. Vous pouvez indifféremment lui demander de :

- faire en sorte qu'une image ait une largeur donnée, il adaptera la hauteur afin de conserver ses proportions ;
- faire en sorte qu'une image ait une hauteur donnée, il adaptera la largeur de la même façon ;
- fixer à la fois la hauteur et la largeur, quitte à déformer l'image ;
- choisir un coefficient de proportionnalité permettant de réduire de façon cohérente et simultanée la hauteur et la largeur de l'image.

La largeur et la hauteur sont fixées respectivement par les variables `width` et `height`. L'échelle, quant à elle, se fixe grâce à la variable `scale` et un coefficient. Ce dernier vaut 1 pour la taille par défaut de l'image, à vous de le diminuer ou de l'augmenter selon vos désirs. Voici plusieurs exemples de codes insérant une image redimensionnée.

```
\includegraphics[width=200]{poulpy.png}
\includegraphics[height=200]{poulpy.png}
\includegraphics[height=200, width=600]{poulpy.png}
% Ici, Poulpy est un peu plate
\includegraphics[scale=1.5]{poulpy.png}
% Ici, Poulpy est plutôt grande
```

Rotation d'une image

La variable `angle` nous permet de faire tourner une image dans un document. Le rendu sur des images rectangulaires n'est pas très esthétique, mais cela peut être utile avec des images ayant une autre forme. La figure 10.4 vous montre l'utilisation de cette option.

```
\includegraphics[angle=45]{poulpy.png} % Poulpy en biais
```

Insertion d'une portion d'image

Il est souvent agréable de pouvoir ajouter uniquement une portion de l'image originale dans un document sans avoir à passer par un logiciel d'édition. Nous allons ici apprendre à isoler une portion rectangulaire d'une image de grande taille.



Les subtilités développées dans ce paragraphe ne fonctionnent (facilement) qu'avec les fichiers PostScript.

Pour notre exemple, nous allons nous exercer sur une photo de Wonder Poulpy et Monsieur Poule que vous voyez en figure 10.5. Vous pouvez la télécharger au format EPS grâce au code web suivant :

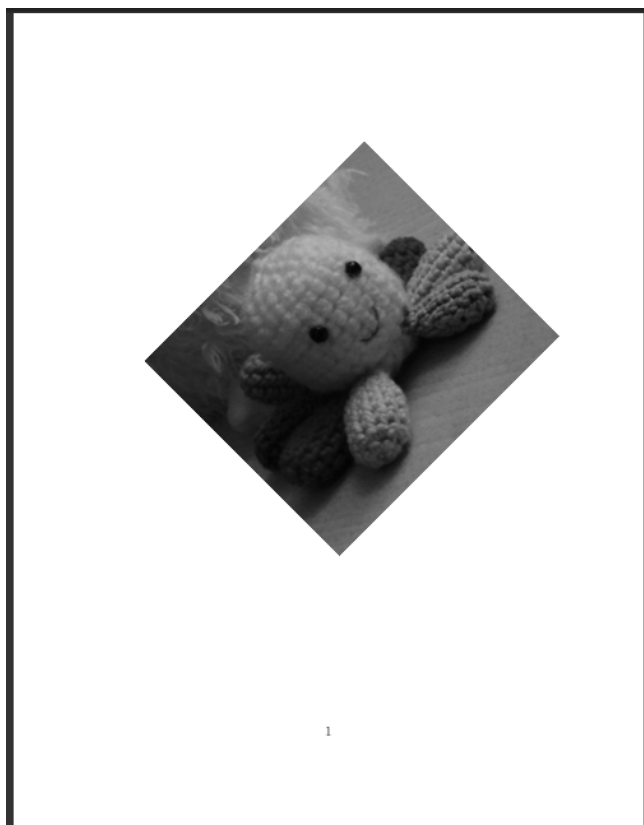


FIGURE 10.4 – Poulpy en biais

▷ Télécharger l'image (EPS)
Code web : 210573



FIGURE 10.5 – Wonder Poulpy et Monsieur Poule

Ici, notre objectif va être d'isoler Poulpy, afin de pouvoir insérer sa photo dans un nouveau document sans son compagnon à plumes. Chaque pixel de l'image représente un point. À chaque point sont associées une abscisse et une ordonnée, l'origine se trouvant sur le coin inférieur gauche de l'image. Si nous souhaitons isoler Poulpy, notre photo de départ devient similaire à celle visible en figure 10.6. La commande à utiliser afin de découper convenablement un rectangle dans une image est la suivante :

```
\includegraphics*[abscisse du coin inférieur gauche,  
↪ ordonnée du coin inférieur gauche][abscisse du coin supérieur droit,  
↪ ordonnée du coin supérieur droit du rectangle]{chemin de l'image}
```

Tentez d'isoler Poulpy ! Je vais tout de même donner la solution à ceux qui n'auraient pas le courage de faire l'exercice (il est un peu difficile pour les moins expérimentés).

```
\includegraphics*[120,20][400,251]{poulpy_et_mr_poule.eps}
```

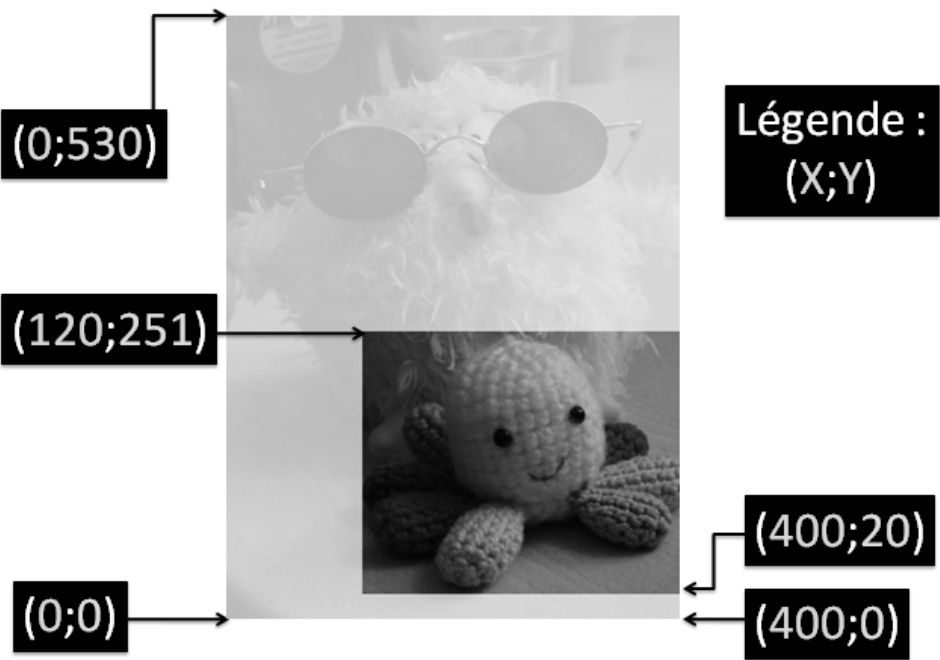


FIGURE 10.6 – Plan de découpage

Intégration d'une image dans un paragraphe

Nous allons maintenant nous pencher sur l'intégration d'une image dans du texte de façon à ce que le texte contourne la figure.



Soyons francs, $\text{\LaTeX}$ n'aime pas trop ça et peut de temps en temps gérer les choses de manière extrêmement exotique.

Le package utilisé ici est `wrapfig`. Il est nécessaire de l'appeler lorsque vous souhaitez inclure l'environnement `wrapfigure` servant à encadrer des images ou des figures par le texte d'un paragraphe. Plusieurs variables font leur apparition avec l'utilisation de l'environnement `wrapfigure` :

- le nombre de lignes nécessaires à la bonne intégration de l'image;
- la taille du dépassement autorisé dans la marge (nous laisserons cette variable à 0 afin de garder des publications propres);
- la largeur de l'image;
- l'alignement de l'image.

La syntaxe à observer est la suivante :

```
\begin{wrapfigure}[nombre de lignes]{placement}{largeur de l'image en cm}
\includegraphics[width=largeur en cm]{votre image}
\end{wrapfigure}
Votre paragraphe, sans saut de ligne entre \end et le paragraphe
```

Le placement se définit à partir des lettres :

- `l` pour placer l'image à gauche;
- `r` pour placer l'image à droite;
- `o` pour placer l'image à l'extérieur, c'est-à-dire à droite pour une page impaire et à gauche pour une page paire;
- `i` pour placer l'image à l'intérieur, c'est-à-dire à gauche pour une page impaire et à droite pour une page paire.

Petit exercice : reprenez l'image de Poulpy donnée en début de chapitre et tentez de l'insérer dans un paragraphe de faux texte. L'image devra faire 4 cm de large, occuper 8 lignes et se trouver à droite de votre paragraphe.

La solution de l'exercice se trouve ci-dessous, illustrée avec la figure 10.7.

```
\documentclass[12pt]{report}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}

\usepackage{wrapfig}
\usepackage{graphicx}
```

```
\begin{document}

\begin{wrapfigure}[8]{r}{4cm}
\includegraphics[width=4cm]{poulpy.png}
\end{wrapfigure}
Gros paragraphe.
\end{document}
```

▷ Copier ce code
Code web : 938332

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut sit amet lectus a odio condimentum porttitor ac cursus orci. Aenean at sapien turpis. Fusce sollicitudin dictum tellus placerat porta. Curabitur lacinia consequat quam. Cras dapibus, sem vitae posuere facilisis, turpis sem facilisis arcu, quis ornare urna risus quis justo. Nunc sagittis blandit lectus sit amet ultrices. Curabitur lacinia consequat quam. Cras dapibus, sem vitae posuere facilisis, turpis sem facilisis arcu, quis ornare urna risus quis justo. Nunc sagittis blandit lectus sit amet ultrices.



FIGURE 10.7 – Rendu de l’insertion

Les flottants

Les techniques pour insérer des images développées dans les paragraphes précédents sont des solutions simples, mais malheureusement peu utilisées. $\text{\LaTeX}$ propose aux auteurs d’optimiser le placement des images et des figures à leur place⁴.

Afin de laisser à $\text{\LaTeX}$ la possibilité de positionner des images à sa guise, nous allons placer ces images dans des environnements dits « flottants ». Le contenu de ces environnements sera placé de la manière que $\text{\LaTeX}$ estimera la plus judicieuse dans votre mise en page.

Bien entendu, vous pourrez indiquer à $\text{\LaTeX}$ votre préférence pour un type de placement en particulier.

Après avoir vu comment placer des flottants insérant des images, nous apprendrons à les légender. Ces légendes seront également traitées ultérieurement.

4. C’est l’une de ses fonctions phares.

L'environnement figure

La création d'un flottant

Pour mettre une image en flottant, nous allons tout simplement insérer la commande habituelle `\includegraphics` dans l'environnement `figure`. Afin de rendre le tout plus élégant dans le document final, nous centrerons l'image⁵ (figure 10.8).

```
\begin{figure}

\begin{center}
\includegraphics{poulpy.png}
\end{center}

\end{figure}
```

Le placement

Il est possible de spécifier à l'environnement `figure` (et par extension à `LATEX`) votre préférence en termes de placement.

Vous pouvez souhaiter que votre flottant soit plutôt :

- placé en haut de page (option `t`) ;
- placé en bas de page (option `b`) ;
- sur une page ne comportant que des flottants (option `p`).

Il existe encore deux options. Vous pouvez indiquer à `LATEX` que vous souhaitez faire figurer l'image dans la zone où vous avez tapé la commande de manière informative (option `h`) ou bien de manière insistante (option `H`).

`H` et `h` sont des outils intéressants, mais que je vous conseille d'employer avec modération. Laissez `LATEX` faire son travail avant de lui imposer des contraintes.

Les options de placement sont spécifiées dans la commande d'ouverture de l'environnement `figure` selon la syntaxe que voici.

```
\begin{figure}[les options non séparées par des virgules]
Quelques démonstrations ci-dessous.
Le « ! » est utilisé ici pour faire comprendre
à LaTeX que nous insistons « énormément » sur une option.
\begin{figure}[b] %nous voulons le flottant en bas.
\begin{figure}[!b] %nous voulons le flottant en bas (avec insistance).
\begin{figure}[bt] %nous voulons le flottant en bas, ou en haut s'il ne peut pas
↪ être en bas.
\begin{figure}[h] %nous voulons le flottant ici.
```

5. Notons que l'environnement `center` utilisé ici n'est pas un simple élément facultatif de mise en page : il nous permettra par la suite de rendre cohérents les alignements des images (et des figures) avec les alignements des légendes.



1

FIGURE 10.8 – Un flottant

```
\begin{figure}[H] %nous voulons le flottant ICI !  
\begin{figure}[hb] %nous voulons le flottant ici, ou en bas si cela n'est pas  
↪ possible.
```

Les placements par défaut

L^AT_EX, selon votre distribution et votre configuration, place les flottants par défaut suivant l'une ou l'autre des options développées ci-dessus. La commande `\floatplacement{type de flottant}{options}` s'utilise afin de changer le comportement par défaut de L^AT_EX.

Dans notre cas, le type de flottant est `figure` et la commande s'utilise par exemple comme ceci :

```
\floatplacement{figure}{t}
```



Cette commande est fournie par le package `float`.

Les légendes

Légender les figures et les images est indispensable en L^AT_EX. Nous allons à présent voir comment ajouter des légendes à nos flottants. C'est le rôle de la commande `\caption{ma légende}`. Elle s'utilise à la suite de l'environnement `center` et précède une éventuelle commande `\label`.

Un démonstration s'impose (figure 10.9) :

```
\begin{figure}  
\begin{center}  
\includegraphics{poulpy.png}  
\end{center}  
\caption{Poulpy est multicolore}  
\label{Poulpy est multicolore}  
\end{figure}
```

Sauts de page

Nous avons appris dans les chapitres traitant de la mise en page à dire à L^AT_EX d'effectuer un saut de page, grâce à la commande `\newpage`. Avec les flottants, deux nouvelles commandes apparaissent :

- `\clearpage`, qui réalise un saut de page tout en imposant à L^AT_EX de mettre tous les flottants en page (il les traitera tous, et produira des pages remplies par les flottants non traités) ;



FIGURE 1 – Poulpy est multicolore

FIGURE 10.9 – Figure légendée

- `\cleardoublepage`, qui a le même effet, si ce n'est qu'il reprend la nouvelle page sur une page impaire.

Comprenez bien l'utilité de ces commandes. Si vous écrivez un rapport en deux grandes parties, il est primordial qu'aucune image de la première partie ne se retrouve dans la seconde à cause d'un choix inopportun de $\text{\LaTeX}$. Ces deux commandes s'utilisent exactement de la même manière que `\newpage`.

En résumé

- Il faut choisir entre l'élaboration d'un document comprenant des figures (PS et EPS) ou des images (JPEG, PNG, BMP...).
- Pour insérer des images, il faut utiliser le package `graphicx` et faire ensuite appel à la commande `\includegraphics`.
- Il est possible d'insérer précisément des images ou figures n'importe où dans un document, mais il est fortement conseillé de laisser $\text{\LaTeX}$ choisir à notre place leur emplacement. On utilise pour cela l'environnement `figure` qui rend l'image flottante : $\text{\LaTeX}$ sélectionne alors l'emplacement le plus approprié pour l'image⁶.
- Pour afficher la légende d'une image, on utilise la commande `\caption`.

6. C'est parfois sur la page suivante!

Chapitre 11

Les tableaux

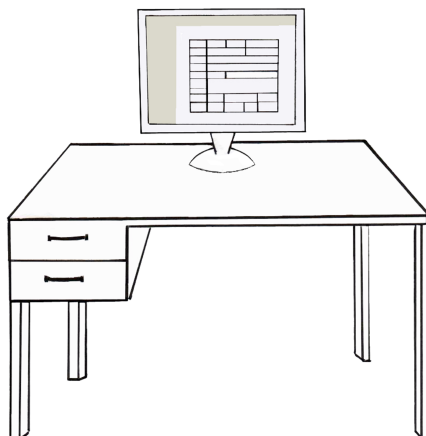
Difficulté : >>>

En sciences, les tableaux sont des éléments essentiels. Les créateurs de $\text{\LaTeX}$ ont su les mettre à l'honneur et apporter une somme colossale de packages et d'environnements sachant les gérer. Massifs et encombrants, comme des images ou des figures, les tableaux sont des flottants et $\text{\LaTeX}$ peut les intégrer comme tels.

Les tableaux, tout comme les notations mathématiques, font l'objet d'une documentation extrêmement fournie. Ce chapitre traitera plusieurs points utiles à la création des tableaux en vous apprenant à :

- composer des tableaux simples ;
- fusionner des cellules ;
- paramétrer le placement des flottants ainsi que quelques autres détails de mise en page.

Bien que difficile, ce chapitre sera extrêmement riche en exemples.



Structure type d'un tableau

Dans cette première partie, nous allons faire nos premiers pas avec des tableaux. Notre but sera de réaliser une grande table de multiplication.

La théorie

Tableau sans bordure

Le tableau sans bordure est le plus simple à créer. Ici, tout se passe dans l'environnement `tabular`, et nous souhaitons créer le tableau visible en figure 11.1. Sa création va se dérouler en trois temps.

1	2
3	4

FIGURE 11.1 – Tableau simple

Dans un premier temps, nous devons décider de l'alignement des cellules dans chaque colonne. Elles peuvent être...

- à droite (option `r`);
- à gauche (option `l`);
- centrées (option `c`).

Dans un deuxième temps, l'une des trois options sera choisie pour chaque colonne et insérée entre accolades à l'ouverture de l'environnement `tabular`.

Ici, nous souhaitons créer un tableau contenant deux colonnes centrées. Notre squelette de départ sera donc le suivant :

```
\begin{tabular}{cc}  
\end{tabular}
```

Pour finir, nous rentrons ligne par ligne le contenu des cellules, séparé par le caractère « `&` ». Chaque ligne se terminera par `\\` (c'est un moyen d'indiquer à $\text{\LaTeX}$ que l'on veut changer de ligne).

Voici donc le code source de notre tableau simple :

```
\begin{tabular}{cc}  
1 & 2 \\  
3 & 4 \\  
\end{tabular}
```



Bien que peu intuitif au début, l'environnement tabular est très pratique.

Tableau avec bordures

Nous souhaitons maintenant ajouter des bordures visibles. Afin de faire apparaître les lignes, il faut placer la commande `\hline` entre deux lignes consécutives, et plus généralement partout où vous souhaitez obtenir une ligne horizontale (dans un tableau).

Démonstration :

```
\begin{tabular}{cc}
\hline
1 & 2 \\
\hline
3 & 4 \\
\hline
\end{tabular}
```

▷ Copier ce code
Code web : 343142

1	2
3	4

FIGURE 11.2 – Les bordures, première étape

Les colonnes, quant à elles, sont un peu plus subtiles à mettre en place. Pour marquer la séparation entre deux colonnes consécutives par une bordure verticale, il faut insérer le caractère « `|` »¹ entre elles lors de la spécification des alignements (figure 11.3).

Ici, `{cc}` devient `{|c|c|}`.

Démonstration :

```
\begin{tabular}{|c|c|}
\hline
1 & 2 \\
\hline
3 & 4 \\
\hline
\end{tabular}
```

1. Le caractère « `|` » est présent en haut de votre clavier. Sur un clavier français, vous pouvez l'insérer avec la combinaison de touches `AltGr + 6`.

▷ Copier ce code
Code web : 725857

1	2
3	4

FIGURE 11.3 – Bordures, seconde étape

Exercice

Il est temps pour vous de souffrir. Dans cet exercice, vous devez reproduire en L^AT_EX l'afreux tableau ci-dessous (figure 11.4). Prenez le temps de réfléchir, il n'y a pas de piège. Recopier ce tableau devrait vous prendre moins de dix minutes.

Multiplié par	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

FIGURE 11.4 – Table de multiplication

La solution :

```
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|}
\hline
Multiplié par & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
\hline
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
\hline
2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
\hline
\end{tabular}
```

```

\hline
3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 \\
\hline
4 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 40 \\
\hline
5 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 \\
\hline
6 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\
\hline
7 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 & 70 \\
\hline
8 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 & 80 \\
\hline
9 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 & 90 \\
\hline
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\
\hline
\end{tabular}

```

▷ Copier ce code
Code web : 756726

Fusion de cellules

Nous allons apprendre à fusionner des lignes et des colonnes. Les commandes exposées ici ne sont pas faciles à prendre en main et demandent un peu d'entraînement et de gymnastique intellectuelle².

Fusion de colonnes

La commande de fusion de colonnes est la suivante :

```

\multicolumn{nombre de colonnes fusionnées}{c, l ou r accompagné
↪ des bordures éventuelles}{le texte de la cellule}

```

La difficulté réside dans la possibilité de choisir de nouveau un alignement pour la cellule fusionnée. Faites bien attention à ce que l'alignement et les bordures de votre cellule soient cohérents avec celles du reste du tableau.

La figure 11.5 met en pratique la commande de fusion de colonnes (regardez bien les différentes syntaxes utilisées) :

```

\begin{tabular}{|c|c|c|c|c|}
\hline

```

2. La difficulté réside dans la mémorisation des commandes. Une fois que vous les connaîtrez par cœur, les choses seront beaucoup plus faciles.

```
1 & \multicolumn{2}{c|}{2} & 3 & 4 \\  
\hline  
\multicolumn{2}{|c|}{5} & 6 & 7 & 8 \\  
\hline  
9 & 10 & \multicolumn{3}{c|}{11} \\  
\hline  
\end{tabular}
```

▷ Copier ce code
Code web : 548701

1	2	3	4
5	6	7	8
9	10	11	

FIGURE 11.5 – Fusion de colonnes

Fusion de lignes

Pour fusionner deux lignes entre elles, on utilise la commande `\multirow`, contenue dans le package du même nom. Elle est dépendante de trois paramètres³ :

```
\multirow{nombre de lignes fusionnées}{taille de la colonne en cm}{votre texte}  
\multirow{nombre de lignes fusionnées}*{votre texte}
```

Vous pouvez voir ci-dessous la démonstration de l'utilisation de `\multirow`. Notez que la commande `\cline` a été introduite. Elle vous permet selon vos souhaits de ne mettre une ligne horizontale qu'entre deux colonnes choisies par vous⁴. Prenez le temps de regarder le code et la figure 11.6 afin de mieux comprendre l'utilité de `\cline`.

```
\begin{tabular}{|l|c|c|c|c|}  
\hline  
1 & \multicolumn{2}{c|}{2} & 3 & 4 \\  
\hline  
\multicolumn{2}{|l|}{5} & 6 & 7 & 8 \\  
\hline  
9 & 10 & \multicolumn{3}{c|}{11} \\  
\hline  
\multirow{2}{1cm}{12} & 13 & 14 & 15 & 16 \\  
\cline{2-5}  
& 17 & 18 & 19 & 20 \\  
\hline
```

3. L'un des paramètres est remplaçable par une étoile.

4. Les colonnes sont numérotées de la gauche vers la droite.

```

21 & 22 & \multirow{2}*{23} & 24 & 25 \\
\cline{1-2} \cline{4-5}
26 & 27 & & 28 & 29 \\
\hline
\end{tabular}

```

▷ Copier ce code
Code web : 338203

1	2	3	4
5	6	7	8
9	10	11	
12	13	14	15
	17	18	19
21	22	23	24
26	27	28	29

FIGURE 11.6 – Fusion de lignes et de colonnes

Autres paramètres applicables à un tableau

De nombreuses options de personnalisation sont disponibles sous L^AT_EX. Nous allons ici étudier les plus courantes avant d'apprendre à nous servir de l'environnement `table`, permettant de faire flotter nos tableaux. Nous les rendrons plus élégants et plus lisibles.

Modifications mineures

Colonne de largeur paramétrée

Nous avons appris précédemment à créer des colonnes en spécifiant simplement l'alignement de leur contenu. Il existe des alternatives au trio `r`, `c` et `l`. Je vais vous en présenter trois.

La nouvelle instruction possible est :

```
|p{largeur de la colonne en centimètres}
```

À la différence de `r`, `c` et `l`, cette instruction n'a pas d'influence sur l'alignement horizontal du texte au sein des cellules. La figure 11.7 illustre son fonctionnement.

```
\begin{tabular}{|p{1cm}|p{2cm}|p{3cm}|p{4cm}|}  
\hline  
1cm & 2cm & 3cm & 4cm \\  
\hline  
\end{tabular}
```

▷ Copier ce code
Code web : 109809

1cm	2cm	3cm	4cm
-----	-----	-----	-----

FIGURE 11.7 – Cellules de longueur définie

Nous apprendrons plus loin dans ce chapitre à utiliser des environnements. Cela vous permettra de choisir l’alignement du contenu dans des colonnes à largeur fixe.

Créer une slashbox

Le package `slashbox` permet d’utiliser la commande `\backslashslashbox{Texte dessous}{Texte dessus}`. Cette commande sert à scinder en deux parties triangulaires de même aire une cellule initialement rectangulaire (figure 11.8).

Démonstration :

```
\begin{tabular}{|c|p{1cm}|p{2cm}|}  
\hline  
\backslashslashbox{Patate}{Carotte} & 1cm & 2cm \\  
\hline  
\end{tabular}
```

▷ Copier ce code
Code web : 649606

Patate	Carotte	1cm	2cm
--------	---------	-----	-----

FIGURE 11.8 – Le package `slashbox`

Changer les séparateurs

Le séparateur habituel lors de la création d’un tableau en $\text{\LaTeX}$ est le caractère « `|` ». Il symbolise la création d’une ligne verticale. Il est néanmoins possible d’utiliser de

nouveaux séparateurs via les commandes `!\{votre séparateur\}` ou `@{votre séparateur}`. Ces commandes sont contenues dans le package `array` (comme beaucoup de commandes utiles à la création de tableaux).

La commande `!\{}` a l'avantage, contrairement à `@{}`, d'insérer une espace avant et après le séparateur. Visuellement, la différence est flagrante (figure 11.9) :

```
\begin{tabular}{|c !{bah} c @{bah}c|}
\hline
texte & texte & texte \\
\hline
\end{tabular}
```

▷ Copier ce code
Code web : 425347

texte bah texte	bahtexte
-----------------	----------

FIGURE 11.9 – Exemple d'utilisation de `!\{}` et `@{}`

Exercice

Tentez de recréer vous-même le tableau mis en avant sur la figure 11.10 (le symbole euro est disponible via la commande `\euro{}`, du package `eurosym`). Il vous faudra bien sûr utiliser les commandes qui viennent d'être introduites.

bonbon	0,50 €
table	14,99 €

FIGURE 11.10 – Forme de la solution

Solution :

```
\begin{tabular}{|c| c @{,} c !{\euro{}}|}
\hline
bonbon & 0 & 50 \\
table & 14 & 99 \\
\hline
\end{tabular}
```

▷ Copier ce code
Code web : 741600

Utilisation de commandes et d'environnements dans un tableau

La conception de tableaux telle que nous venons de l'apprendre ne permet pas l'insertion de commandes ou même d'environnements dans les colonnes. Avouons-le, c'est fâcheux. Pour pouvoir agir sur les colonnes, il vous faudra utiliser `<{votre commande ou votre \begin{..} }` avant l'instruction de création de colonne (1, r, c ou `\p{}`) et `>{votre commande ou votre \end{..} }` juste après si vous le souhaitez.

Démonstration avec la table de multiplication que nous avons utilisée en début de chapitre (figure 11.11) :

```
\begin{tabular}{|>{\begin{bf}} c <{\end{bf}}|c|c|c|c|c|c|c|c|c|}
\hline
Multiplié par & \begin{bf}1\end{bf} & \begin{bf}2\end{bf} & \begin{bf}3\end{bf} & \begin{bf}4\end{bf} & \begin{bf}5\end{bf} & \begin{bf}6\end{bf} & \begin{bf}7\end{bf} & \begin{bf}8\end{bf} & \begin{bf}9\end{bf} & \begin{bf}10\end{bf} \\
\hline
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
\hline
2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 \\
\hline
3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 \\
\hline
4 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 40 \\
\hline
5 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 \\
\hline
6 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\
\hline
7 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 & 70 \\
\hline
8 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 & 80 \\
\hline
9 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 & 90 \\
\hline
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\
\hline
\end{tabular}
```

▷ Copier ce code
Code web : 445484

Multiplié par	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

FIGURE 11.11 – Mise en gras de certaines parties

Colorer des cellules

Nous allons ici apprendre à colorer des lignes, des cellules ainsi que des colonnes. Deux packages sont à utiliser : `color` (dont nous nous sommes déjà servi) et `colortbl`.

Les commandes sont les suivantes :

- `\columncolor{couleur}` pour colorer les colonnes ;
- `\rowcolor{couleur}` pour colorer les lignes ;
- `\cellcolor{couleur}` pour colorer les cellules.

Exercice

Réalisez le tableau visible sur la figure 11.12, en colorant la première ligne et la première colonne sur fond jaune⁵ :

Solution :

```
\begin{tabular}{>{\begin{bf} \columncolor{yellow}} c <{\end{bf}}ccccccccc}
\rowcolor{yellow}Multiplié par & \begin{bf}1\end{bf} & \begin{bf}2\end{bf} &
↪ & \begin{bf}3\end{bf} & \begin{bf}4\end{bf} & \begin{bf}5\end{bf} &
↪ & \begin{bf}6\end{bf} & \begin{bf}7\end{bf} & \begin{bf}8\end{bf} &
↪ & \begin{bf}9\end{bf} & \begin{bf}10\end{bf} & \\
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & \\
2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 & \\
3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 & \end{tabular}
```

5. Vous avez le droit de vous servir des codes précédemment vus dans le chapitre.

Multiplié par	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

FIGURE 11.12 – Tableau à reproduire

```
4 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 40 \\
5 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 \\
6 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\
7 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 & 70 \\
8 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 & 80 \\
9 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 & 90 \\
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\
\end{tabular}
```

► Copier ce code
Code web : 270222



Vous constaterez que les tableaux ne sont pas difficiles à réaliser mais demandent de la rigueur lors de la conception.

Un environnement flottant : table

Faire flotter un tableau revient à insérer l’environnement `tabular` dans un nouvel environnement qui, lui, est flottant. Celui-ci s’appelle `table` et son fonctionnement interne est similaire à celui de l’environnement `figure` vu précédemment, avec `caption`,

`label`, `center`, etc. Rien ne diffère dans leur fonctionnement.

Tentez de faire **flotter** le tableau vu dans le paragraphe précédent tout en écrivant en légende « Tables de multiplication ». Rappelez-vous de la syntaxe ci-dessous et tout ira bien (figure 11.13).

```
\begin{table}
\begin{center}
\begin{tabular}
Votre tableau
\end{tabular}
\end{center}
\caption{votre légende}
\label{votre référence}
\end{table}
```

Multiplié par	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

TABLE 1 – Tables de multiplication

FIGURE 11.13 – Tableau flottant

La solution :

```
\begin{table}[h]
\begin{center}
\begin{tabular}{>{\begin{bf} \columncolor{yellow}} c <{\end{bf}}ccccccccc}

\rowcolor{yellow}Multiplié par & \begin{bf}1\end{bf} & \begin{bf}2\end{bf} \\
↪ & \begin{bf}3\end{bf} & \begin{bf}4\end{bf} & \begin{bf}5\end{bf} \\
↪ & \begin{bf}6\end{bf} & \begin{bf}7\end{bf} & \begin{bf}8\end{bf} \\
↪ & \begin{bf}9\end{bf} & \begin{bf}10\end{bf} & \\

1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & \\

2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20 & \\

3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 & \end{tabular}
\end{center}
\end{table}
```

```
4 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 40 \\
5 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 \\
6 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\
7 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 & 70 \\
8 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 & 80 \\
9 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 & 90 \\
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\
\\end{tabular}
\\end{center}
\\caption{Tables de multiplication}
\\label{mes belles tables}
\\end{table}
```

▷ Copier ce code
Code web : 588979

En résumé

- Un tableau nécessite un formalisme et un code propre.
- Un tableau se crée dans l’environnement `tabular`.
- Les cellules d’une même ligne sont séparées par le symbole `&`. Pour passer à la ligne suivante, on insère un retour à la ligne avec `\\`.
- Il est possible d’utiliser les environnements et les commandes de mises en forme de texte au sein d’un tableau.
- Pour rendre un tableau flottant, et laisser $\text{\LaTeX}$ gérer sa position, il faut l’entourer de l’environnement `table`.

Chapitre 12

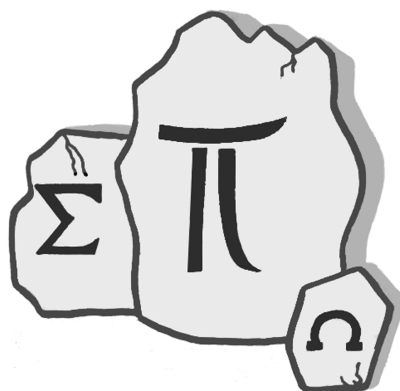
Les mathématiques

Difficulté : 

Vous venez de survivre aux tableaux, c'était le chapitre le plus difficile de ce cours. Il est maintenant temps d'apprendre à écrire des expressions mathématiques¹, des formules, des intégrales.

Nous traiterons dans l'ordre :

- des manières d'introduire des maths dans un document ;
- des équations simples ;
- des systèmes d'équations, ainsi que de tout ce qui ressemble de près ou de loin à une accolade ;
- des matrices et des théorèmes ;
- des flèches et symboles étirables.



1. C'est le domaine de prédilection de $\text{\LaTeX}$.

Il va de soi que certains d'entre vous n'ont pas les connaissances mathématiques ou physiques nécessaires pour saisir l'utilité d'une partie des outils mathématiques (discriminants, matrices, exponentielles...) et des notations qui leur sont associées. Ne prenez pas peur, ceci n'est pas un cours de maths, mais un cours destiné à vous apprendre à écrire des expressions mathématiques. Il vous suffit de sauter les paragraphes qui ne vous sont pas utiles.

Saisir une expression mathématique

Cette première partie présente les éléments vous permettant d'insérer des expressions mathématiques ou scientifiques au sein d'un texte. Ces commandes et environnements ont pour effet de passer de la police standard du document² à la police définie pour les maths³.

Ce changement de police est aussi accompagné d'un changement global des règles de typographie. Par exemple, il n'y aura pas la même espace de chaque côté du signe « — » selon que vous êtes dans l'environnement de texte ou dans celui de l'écriture scientifique.

Deux possibilités vous sont offertes par L^AT_EX. Vous pouvez choisir d'écrire des expressions scientifiques au milieu de votre texte, ou bien le faire à part, dans un espace où elles seront plus lisibles.

Bien sûr, ce serait trop simple s'il n'y avait qu'une seule méthode, nous allons donc en examiner plusieurs tout au long de ce chapitre.

- Pour écrire des expressions mathématiques au sein d'un texte⁴, il faudra entourer les expressions par des `$`, ou utiliser l'environnement `math`.
- Pour écrire des expressions mathématiques en dehors d'un paragraphe, il faudra les entourer par les signes `\[` (à leur commencement) et `\]` (à leur clôture)⁵, ou utiliser l'environnement `equation`⁶.
- Pour écrire du texte dans une expression mathématique, il faudra utiliser la commande `\text{mon bout de texte}`.

N'oublions pas que la bonne lisibilité d'un document met de bonne humeur son lecteur. Nous privilégierons donc la seconde solution : écrire les expressions scientifiques en dehors des paragraphes.

Une démonstration de toutes ces commandes se trouve ci-dessous. Les commandes et environnements sont volontairement noyés dans des paragraphes afin que vous puissiez mieux voir leurs effets (figure 12.1).

2. Nous avons appris à la modifier dans le chapitre traitant des polices.

3. Il est également possible de la modifier grâce aux packs de polices.

4. Beaucoup de commandes ne donnent pas le même résultat lorsqu'elles sont utilisées au sein d'un texte. Ce n'est pas une erreur de L^AT_EX mais une optimisation. Ne vous étonnez donc pas si le rendu d'une formule mathématique est parfois différent lorsque celle-ci est insérée dans un paragraphe et non en dehors.

5. Il est aussi possible de remplacer chacun de ces signes par `$$`.

6. Plutôt spécifique aux équations, il permet de numéroté ces dernières.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt
↪ pharetra,  $x+y+z=4$  nisi sapien ornare felis, ut tristique tellus
↪ justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt
↪ pharetra,  $[x+y+z=4]$  nisi sapien ornare felis, ut tristique
↪ tellus justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt
↪ pharetra,  $\begin{math}x+y+z=4\end{math}$  nisi sapien ornare felis,
↪ ut tristique tellus justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt
↪ pharetra,  $\begin{equation}x+y+z=4\end{equation}$  nisi sapien ornare felis,
↪ ut tristique tellus justo id ipsum.

```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non pharetra, $x + y + z = 4$ nisi sapien ornare felis, ut tristique tellus justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt pharetra,

$$x + y + z = 4$$

nisi sapien ornare felis, ut tristique tellus justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt pharetra, $x + y + z = 4$ nisi sapien ornare felis, ut tristique tellus justo id ipsum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vitae faucibus augue velit ac elit. Aliquam ullamcorper, lectus non tincidunt pharetra,

$$x + y + z = 4 \tag{1}$$

nisi sapien ornare felis, ut tristique tellus justo id ipsum.

FIGURE 12.1 – Insertion d'équations sommaires

Ces commandes et environnements sont des incontournables et ils savent montrer leur utilité dans de nombreux cas. Les pages suivantes vont introduire un grand nombre d'exemples et d'objets mathématiques.

Fonctions usuelles

Cette partie traite de fonctions assez faciles à écrire. Si quelques-unes d'entre elles sont des commandes qui prennent des arguments (mais n'utilisent pas forcément le `\` habituel), vous pourrez constater que d'autres s'introduisent exactement de la même manière qu'un caractère spécial, comme le cosinus par exemple (*cos* pour les intimes).

Fonctions usuelles à arguments

Nous allons commencer en douceur, avec ci-dessous une série de fonctions qui prennent des arguments, chacune d'elles étant accompagnée d'une capture et d'un exemple. Essayez de les utiliser, elles ne mordent pas.

Fraction

`| \[\frac{numérateur}{dénominateur} \]`

▷ Copier ce code
Code web : 662456

$$\frac{numérateur}{dénominateur}$$

Fractions en cascade (package `amsmath`)

`| \[\cfrac{num1}{den1} + \cfrac{num2}{den2} \]`

▷ Copier ce code
Code web : 328183

$$\frac{num1}{den1 + \frac{num2}{den2}}$$

Racine

`| \[\sqrt[15]{nombre} \]`

▷ Copier ce code
Code web : 213956

$$\sqrt[15]{\text{nombre}}$$

Limite

`| \[ \lim_{\text{nbr1} \rightarrow \text{nbr2}} x+42 \]`

▷ Copier ce code
Code web : 815893

$$\lim_{\text{nbr1} \rightarrow \text{nbr2}} x + 42$$

Exposant

`| \[ \text{nombre}^{\text{exposant}} \]`

▷ Copier ce code
Code web : 696036

$$\text{nombre}^{\text{exposant}}$$

Indice

`| \[ \text{nombre}_{\text{indice}} \]`

▷ Copier ce code
Code web : 566699

$$\text{nombre}_{\text{indice}}$$

Fonctions usuelles sans argument

Le tableau suivant contient des commandes s'utilisant comme des caractères spéciaux. Par exemple, plutôt qu'une hypothétique commande `\cos{argument}` pour le cosinus, il y a le caractère spécial `\cos`, qui peut être suivi de ce que vous voulez (un exposant, un θ ou n'importe quoi d'autre). Ci-dessous, une petite démonstration des possibilités que cela offre, suivi d'un gros tableau⁷.

7. Les caractères grecs sont présentés dans l'annexe A.

`\[cos]`
`\[cos^{42}(\lambda)]`

`cos`

`cos42(λ)`

Voilà donc la preuve par l'exemple que l'utilisation du cosinus est alors identique à celle des commandes élémentaires, ce qui rend les choses plus simples, non ?

Fonctions usuelles sans argument			
Nom	Code	Nom	Code
<i>cos</i>	<code>\cos</code>	<i>sin</i>	<code>\sin</code>
<i>arccos</i>	<code>\arccos</code>	<i>arcsin</i>	<code>\arcsin</code>
<i>cosh</i>	<code>\cosh</code>	<i>sinh</i>	<code>\sinh</code>
<i>tan</i>	<code>\tan</code>	<i>arg</i>	<code>\arg</code>
<i>arctan</i>	<code>\arctan</code>	<i>ln</i>	<code>\ln</code>
<i>tanh</i>	<code>\tanh</code>	<i>log</i>	<code>\log</code>
<i>exp</i>	<code>\exp</code>	<i>dim</i>	<code>\dim</code>
<i>min</i>	<code>\min</code>	<i>max</i>	<code>\max</code>

Le tableau que voilà contient des commandes permettant d'écrire du texte en caractères romains plutôt qu'en italique mathématique. En effet, s'il existe une commande fondamentale servant à mettre n'importe quel texte en caractères romains, `\mathrm{votre fonction}`, celle-ci est fastidieuse à utiliser, et de bonnes âmes ont créé des raccourcis tels que `\cos` pour vous éviter d'avoir à écrire `\mathrm{cos}` à chaque cosinus que vous employez. Nous utiliserons la commande `\mathrm{fonction}` lorsqu'il nous faudra écrire du texte ou une fonction ne disposant pas d'un tel raccourci.

Les intégrales

Préambule technique

À partir de ce paragraphe, les choses se compliquent un peu (mais pas beaucoup, rassurez-vous). Vous allez faire cohabiter dans vos expressions des symboles tels que des intégrales, des fonctions et plein d'autres petites choses bien sympathiques. Le problème auquel vous aurez à faire face est qu'à force de tout mélanger, vous finirez par ne plus vraiment savoir quels packages appeler.

Alors, soyons simples et allons-y comme des brutes ! Nous allons tout simplement appeler dès le préambule les trois principaux packages nécessaires à l'écriture d'expressions scientifiques, et nous n'aurons plus à nous soucier du package auquel nous faisons appel

pour utiliser l'un ou l'autre symbole⁸. Ici, je les ai classés par ordre alphabétique, mais cela n'a strictement aucune importance à notre niveau⁹.

```
\usepackage{amsmath}
\usepackage{amssymb}
\usepackage{mathrsfs}
```

Souvent, vous tomberez sur des documentations indiquant des commandes L^AT_EX sans préciser les packages qui les fournissent. Ces trois-là devraient vous sortir de bien des situations désagréables.

Intégrales et expressions plus complexes

Les commandes simples ayant été introduites, nous allons apprendre à construire des expressions un peu plus ambitieuses. Nous allons maintenant nous pencher sur la rédaction de formules comportant des intégrales (simples, doubles, triples et même plus!), des sommes ou des produits.

Un exemple étant beaucoup plus facile à comprendre qu'un long discours, j'ai dressé pour vous une liste vous présentant des formules générales¹⁰ ainsi que des utilisations de celles-ci. Vous serez confronté à la commande `\` suivie d'une espace : elle sert simplement à insérer une espace dans un environnement mathématique.

Intégrale simple

Modèle

```
\[ \int {contenu} \]
\[ \int_{\text{borne inférieure}}^{\text{borne supérieure}} {contenu} \]
\[ \int \limits_{\text{borne inférieure}}^{\text{borne supérieure}} {contenu} \]
```

Démonstration

```
\[ \int {x^2 dx} \]
\[ \int_{1}^3 {x^2 dx} \]
\[ \int \limits_{1}^3 {x^2 dx} \]
```

8. Vous imaginez une table de 300 symboles avec à côté de chacun le nom du package ? Ce serait tout simplement ignoble.

9. Dans une utilisation plus avancée de L^AT_EX, les modifications apportées par les packages sur les commandes peuvent interférer entre elles (deux packages qui définissent la même commande, par exemple). Généralement, le dernier package introduit a le dernier mot. Ici, nous ne nous intéressons pas à ces cas atypiques.

10. Remarquez qu'il existe des commandes dérivées de `\int` (la commande pour le symbole d'intégration) permettant de générer des intégrales doubles ou triples (respectivement, `\iint` et `\iiint`). Celles-ci fonctionnent de la même manière que `\int`.

▷ Copier ce code
Code web : 612269

$$\int x^2 dx$$

$$\int_1^3 x^2 dx$$

$$\int_1^3 x^2 dx$$

Intégrale sur une courbe fermée

Modèle

`\oint {contenu}`

Démonstration

`\oint {x^2\ dx}`

▷ Copier ce code
Code web : 375829

$$\oint x^2 dx$$

Symbole de la somme

Modèle

`\sum_{en dessous}^{au dessus} contenu`

Démonstration

`\sum_{k=2}^{47} k+1`

▷ Copier ce code
Code web : 474034

$$\sum_{k=2}^{47} k + 1$$

Symbole du produit

Modèle

$\prod_{\text{en dessous}}^{\text{au dessus}} \text{contenu}$

Démonstration

$\prod_{k=2}^{47} k+1$

▷ Copier ce code
Code web : 584070

$$\prod_{k=2}^{47} k+1$$

En réalité, les intégrales doubles et triples peuvent également s'écrire avec des intégrales simples à l'intérieur d'autres intégrales simples. Vous trouverez des exemples ci-dessous, illustrant des cas plus ou moins exotiques.

$\int \int \{x^2+y\} \, dx \, dy$
 $\int \int_{\int_1^3 \{x^2+y\} \, dx \, dy}$
 $\int \int \int_{\int_1^3 \{x^2+y\} \, dx \, dy} \{x^2+y\} \, dx \, dy$
 $\int \int_{\int_0^4 \int_1^3 \{x^2+y\} \, dx \, dy} \{x^2+y\} \, dx \, dy$
 $\int \int \int_{\int_0^4 \int_1^3 \{x^2+y+z\} \, dx \, dy \, dz} \{x^2+y+z\} \, dx \, dy \, dz$
 $\int \int_{\int_0^5 \int_0^4 \int_1^3 \{x^2+y+z\} \, dx \, dy \, dz} \{x^2+y+z\} \, dx \, dy \, dz$
 $\int \int \int_{\int_0^5 \int_0^4 \int_1^3 \{x^2+y+z\} \, dx \, dy \, dz} \{x^2+y+z\} \, dx \, dy \, dz$

▷ Copier ce code
Code web : 136808

$$\int \int x^2 + y \, dx \, dy$$

$$\int \int_1^3 x^2 + y \, dx \, dy$$

$$\int \int_1^3 x^2 + y \, dx \, dy$$

$$\int_0^4 \int_1^3 x^2 + y \, dx \, dy$$

$$\int_0^4 \int_1^3 x^2 + y \, dx dy$$

$$\int \int \int x^2 + y + z \, dx dy dz$$

$$\int_{x=0}^{x=5} \int_{y=0}^{y=4} \int_{z=1}^{z=3} x^2 + y + z \, dx dy dz$$

$$\int_{x=0}^{x=5} \int_{y=0}^{y=4} \int_{z=1}^{z=3} x^2 + y + z \, dx dy dz$$

Si le besoin d'écrire plusieurs lignes sous un opérateur se fait sentir, utilisez la commande `\substack{}`. Il faut alors séparer chaque ligne par les habituels « `\` ».

```
\[\sum_{\substack{k=0 \\\ i=0 \\\ j=0}}^n i+j+k\]  
\[\prod_{\substack{k=0 \\\ i=0 \\\ j=0}}^n i+j+k\]
```

▷ Copier ce code
Code web : 560547

$$\sum_{\substack{k=0 \\ i=0 \\ j=0}}^n i + j + k$$

$$\prod_{\substack{k=0 \\ i=0 \\ j=0}}^n i + j + k$$

Les systèmes d'équations

Les systèmes d'équations sont simples à mettre en place : ils requièrent l'utilisation d'environnements ayant des fonctionnalités similaires à celles de `tabular`, si ce n'est qu'ils mettent en forme... des équations.

L'environnement eqnarray

Notre choix va ici se porter sur l'environnement `eqnarray`, qui offre la possibilité de créer un système d'équations proprement aligné, grâce au caractère `&`. Notez que l'environnement `eqnarray` numérote les équations, mais que son homologue `eqnarray*` ne le fait pas.

Démonstration :

```
\begin{eqnarray}
y &=& x - y + z \\
x &=& y \\
z &=& y
\end{eqnarray}

\begin{eqnarray*}
y &=& x - y + z \\
x &=& y \\
z &=& y
\end{eqnarray*}

\begin{eqnarray}
y &=& x - y + z \\
x &=& y \\
z &=& y
\end{eqnarray}
```

▷ Copier ce code
Code web : 121063

$$y = x - y + z \quad (12.1)$$

$$x = y \quad (12.2)$$

$$z = y \quad (12.3)$$

$$y = x - y + z$$

$$x = y$$

$$z = y$$

$$y = x - y + z \quad (12.4)$$

$$x = y \quad (12.5)$$

$$z = y \quad (12.6)$$

Vous ne manquerez pas de remarquer que la numérotation des équations se poursuit le long du document. C'est très pratique pour guider vos lecteurs, et un lecteur heureux est un lecteur qui vous aime. Important, n'est-ce pas ?

L'environnement align

Sans doute aurez-vous souvent recours à l'environnement `eqnarray`. Néanmoins, celui-ci pose quelques problèmes d'alignement et d'espacement dans des cas particuliers. Il est alors plus adapté de choisir l'environnement `align`¹¹. La commande `align` fonctionne de la même manière que la commande `eqnarray` ; de même, vous avez la possibilité d'utiliser `align*`, il y a juste une petite différence dans la syntaxe que vous pouvez remarquer dans le code source ci-dessous.

```
\begin{align}
y &= x - y + z \\
x &= y \\
z &= y \\
\end{align}

\begin{align*}
y &= x - y + z \\
x &= y \\
z &= y \\
\end{align*}

\begin{align}
y &= x - y + z \\
x &= y \\
z &= y \\
\end{align}
```

▷ Copier ce code
Code web : 781484

Les grands acharnés peuvent lire le document vers lequel pointe le code web ci-dessous. Il développe les différences entre `eqnarray` et `align`.

▷ A propos de `eqnarray`
Code web : 353031

L'environnement array

Notre prochaine mission est de positionner, dans un premier temps, des parenthèses autour de notre système d'équations et dans un second temps, une accolade ouvrante

11. Vous vous demandez sûrement pourquoi je vous ai d'abord présenté une notion « bancale ». C'est tout simplement parce que vous la croiserez de façon extrêmement fréquente dans des codes sources L^AT_EX.

à son extrémité gauche. Pour ce faire, nous allons utiliser la syntaxe suivante (nous utilisons ici `array` pour aligner les équations et choisir l'alignement de chaque colonne) :

```
\[
\left code_du_délimiteur_de_gauche
\begin{array}{r c l}
y &=& x - y + z\\
x &=& y\\
z &=& y
\end{array}
\right code_du_délimiteur_de_droite
\]
```

▷ Copier ce code
Code web : 357267

Pour pouvoir installer nos parenthèses et notre accolade, il nous faut jeter un œil au tableau des délimiteurs contenant les codes correspondant à chacun d'entre eux (le point correspond à une absence de délimiteur lors de l'utilisation de `\right` et `\left`).

Vous avez l'exercice, vous avez les codes des délimiteurs... À vous de jouer !

La solution :

```
\[
\left(
\begin{array}{r c l}
y &=& x - y + z\\
x &=& y\\
z &=& y
\end{array}
\right)

\left\{
\begin{array}{r c l}
y &=& x - y + z\\
x &=& y\\
z &=& y
\end{array}
\right.
\]
```

▷ Copier ce code
Code web : 127125

$$\left(\begin{array}{rcl} y & = & x - y + z \\ x & = & y \\ z & = & y \end{array} \right)$$

Les délimiteurs					
Code	Rendu	Code	Rendu	Code	Rendu
(	(	)	)		
[	[	]	]		
\{	{	\}	}	\backslash	\
\uparrow	↑	\downarrow	↓	/	/
\updownarrow	↕	\Uparrow	⇑	\rangle	>
\Downarrow	⇓	\Updownarrow	⇕	\langle	<
\lfloor	⌊	\rfloor	⌋	\lceil	⌈
		\rceil	⌉		

$$\begin{cases} y &= x - y + z \\ x &= y \\ z &= y \end{cases}$$

Les matrices

Vous vous figurez sans doute qu'une matrice — ou l'expression de son déterminant — se conçoit à l'aide de la commande `array`, eh bien c'est raté ! Cela est certes possible, mais des environnements ont été spécialement conçus pour vous aider à écrire des matrices dans vos documents.

Les différents environnements

Vous n'allez pas être dépaycé : la syntaxe est toujours la même que celle que l'on connaît pour les tableaux, la seule chose qui change est le nom des différents environnements. À nouveau, nous allons passer en revue la liste des différentes options qui vous sont proposées.

L'environnement `matrix`

```
\[
\begin{matrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{matrix}
\]
```

▷ Copier ce code
Code web : 517521

1	2	3
4	5	6
7	8	9

L'environnement `pmatrix`

```
\[
\begin{pmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{pmatrix}
\]
```

```
| 7 & 8 & 9
| \end{pmatrix}
| \]
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

L'environnement `vmatrix`

```
| \[
| \begin{vmatrix}
| 1 & 2 & 3 \\
| 4 & 5 & 6 \\
| 7 & 8 & 9
| \end{vmatrix}
| \]
```

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}$$

L'environnement `Vmatrix`

```
| \[
| \begin{Vmatrix}
| 1 & 2 & 3 \\
| 4 & 5 & 6 \\
| 7 & 8 & 9
| \end{Vmatrix}
| \]
```

$$\begin{Vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{Vmatrix}$$

L'environnement `bmatrix`

```
\[
\begin{bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{bmatrix}
\end{bmatrix}
```

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

L'environnement `Bmatrix`

```
\[
\begin{Bmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{Bmatrix}
\end{Bmatrix}
```

$$\begin{Bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{Bmatrix}$$

Exemple pratique

Petite application de l'environnement `matrix` et de ses pairs, voici un exemple de matrice contenant des séries de trois points horizontaux (commande `\cdots`), verticaux (commande `\vdots`) ou diagonaux (commande `\ddots`). Effet garanti ¹².

```
\[
\begin{pmatrix}
a_{11} & \cdots & a_{1k} \\
\vdots & \ddots & \vdots \\
a_{k1} & \cdots & a_{kk}
\end{pmatrix}
\end{pmatrix}
```

12. Vous pouvez vous servir de ces trois commandes dans d'autres contextes si vous le souhaitez.

▷ Copier ce code
Code web : 755791

$$\begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix}$$

La commande `\phantom{}`

Voici une autre petite astuce : la commande `\phantom{texte}`, qui permet d'insérer un espace de la longueur du texte contenu entre ses crochets. Dans le contexte des matrices, elle est très utile, car elle permet d'insérer des blancs et d'aligner les différents éléments. Cette commande n'est cependant pas spécifique aux matrices : elle peut servir dans beaucoup d'autres cas.

```
\[ % Sans \phantom{}
\begin{Vmatrix}
  1 & 12345 & 3 \\
  94 & 5 & -6 \\
  7 & 8 & 9
\end{Vmatrix}
\]

\[ % Avec \phantom{}
\begin{Vmatrix}
  \phantom{9}1 & 12345 & \phantom{-}3 \\
  94 & \phantom{1234}5 & -6 \\
  \phantom{9}7 & \phantom{1234}8 & \phantom{-}9
\end{Vmatrix}
\]
```

▷ Copier ce code
Code web : 578819

$$\left\| \begin{array}{ccc} 1 & 12345 & 3 \\ 94 & 5 & -6 \\ 7 & 8 & 9 \end{array} \right\|$$

$$\left\| \begin{array}{ccc} 1 & 12345 & 3 \\ 94 & 5 & -6 \\ 7 & 8 & 9 \end{array} \right\|$$

Mise en forme de théorèmes, lemmes...

À l'instar des citations, les théorèmes, corollaires, lemmes et autres affreuses petites choses écrites par de vieux barbus doivent respecter leurs conventions typographiques. Nous allons apprendre à mettre tout cela en place (eh non, vos souffrances ne sont pas terminées).

Généralités



Tout d'abord, clarifions un point. Un abus de langage assez affreux consiste à dire qu'on « écrit un théorème » à chaque fois que l'on emploie la syntaxe dont je vais vous parler dans les lignes qui suivent. Si vous êtes physicien, personne ne vous en voudra¹³. En revanche, si vous faites des maths, évitez de dire à tout va « j'écris un théorème » quand vous mettez en page un corollaire : vous éviterez ainsi les multiples railleries de vos fourbes collègues.

Revenons à nos moutons ! Nous cherchons à mettre en page différentes choses : des corollaires, des lois, des théorèmes... Il faut bien sûr que chaque type d'entité ait une numérotation qui lui soit propre au sein du document (si dans une page, vous tapez un théorème et un corollaire, le théorème devra s'appeler « théorème 1 » et le corollaire « corollaire 1 », pas « corollaire 2 »).

Petit plus, nous souhaiterions pouvoir numérotter les différentes entités par rapport aux chapitres, sections et autres éléments structuraux. Ainsi, si un théorème est le troisième du chapitre cinq, il portera le nom de « théorème 5.3 » (même idée pour les sections et les autres éléments de structure).

Tout ceci se fait très facilement grâce à $\text{\LaTeX}$, mais en deux temps. Dans un premier temps, il nous faudra indiquer à $\text{\LaTeX}$ dans le préambule qu'il faut créer les différentes entités utilisant la mise en page dédiée au théorème. Dans le corps du texte, nous pourrons ensuite simplement utiliser la commande permettant d'inclure des théorèmes, des lois et d'autres joyeusetés.

Pratiquons !

Si vous avez lu les paragraphes précédents, vous avez dû garder en tête la méthode en deux temps. La première chose à faire est donc d'appeler dans le préambule la commande `\newtheorem` (déclinée de plusieurs façons) avant d'utiliser un nom d'environnement de votre choix dans la suite du document.

```
\documentclass{book}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

13. La preuve, c'est que je suis physicien, et je ne vous en voudrai pas.

```
\usepackage[français]{babel}

\usepackage{amsthm}
\usepackage{amsmath}
\usepackage{amssymb}
\usepackage{mathrsfs}

\newtheorem*{petit_nom}{Lemme} % La petite étoile enlève la numérotation, mais
↪ nécessite le package amsthm
\newtheorem{petit_nom1}{Proposition}
\newtheorem{petit_nom2}{Définition}[chapter] % Le [chapter] peut par exemple être
↪ remplacé par [section], il permet de numérotter les éléments par rapport aux
↪ numéros de chapitre
\newtheorem{petit_nom3}{Définition}[section]

\begin{document}
\chapter{Les ours}
\chapter{Les lapins}
\section{les lapins nains}

\begin{petit_nom}
ils aiment les carottes
\end{petit_nom}

\begin{petit_nom}[des lapins] % Argument optionnel, j'ajoute le nom de la loi
↪ entre crochets
ils aiment les carottes
\end{petit_nom}

\begin{petit_nom1}
ils aiment les carottes
\end{petit_nom1}

\begin{petit_nom1}[des lapins] % Argument optionnel, j'ajoute le nom entre
↪ crochets
ils aiment les carottes
\end{petit_nom1}

\begin{petit_nom2}
ils aiment les carottes
\end{petit_nom2}

\begin{petit_nom2}[des lapins] % Argument optionnel, j'ajoute le nom entre
↪ crochets
ils aiment les carottes
\end{petit_nom2}

\section{les autres}
```

```
\begin{petit_nom3}
ils aiment les carottes
\end{petit_nom3}

\begin{petit_nom3}[des lapins] % Argument optionnel, j'ajoute le nom entre
↪ crochets
ils aiment les carottes
\end{petit_nom3}

\end{document}
```

▷ Copier ce code
Code web : 427258

L'exemple est très explicite (figure 12.2) : la façon dont vous utiliserez la commande `\newtheorem` est importante, mais il n'y a rien de plus à savoir utiliser. Sachez que dans de nombreux cas, des commandes complexes en $\text{\LaTeX}$ vous seront présentées par le biais d'un code commenté et rien d'autre. Prenez le temps de le décortiquer : vous verrez alors que tout deviendra plus simple (enfin, tout ce qui concerne $\text{\LaTeX}$). C'est la raison pour laquelle tout ce qui est nécessaire à la compréhension de ce concept est intégré au code précédent et ne figure pas dans un paragraphe explicatif.

Flèches, symboles étirables et espaces

Cette avant-dernière partie recense de nombreuses commandes à la fois paramétrables et utiles en $\text{\LaTeX}$. Cela va des différents types d'espaces aux accolades, en passant par les flèches et les accents.

Ici ne sont listées **que** les commandes paramétrables, toutes les autres se trouvent dans la partie mathématique de l'annexe traitant des caractères spéciaux (sinon, ce chapitre deviendrait vraiment trop long).

Les espaces

Différents types d'espaces sont disponibles dans un environnement mathématique. Notez que l'une d'elles est négative et permet de rapprocher des éléments. Un exemple de rendu est montré à la figure 12.3.

```
\[1\!2\]
\[1\,2\]
\[1\ 2\]
\[1\;2\]
\[1\!2\]
\[1\quad2\]
\[1\qquad2\]
```

Chapitre 2

Les Lapins

2.1 les lapins nains

- Lemme.** *ils aiment les carottes*
- Lemme** (des lapins). *ils aiment les carottes*
- Proposition 1.** *ils aiment les carottes*
- Proposition 2** (des lapins). *ils aiment les carottes*
- Définition 2.1.** *ils aiment les carottes*
- Définition 2.2** (des lapins). *ils aiment les carottes*

2.2 les autres

- Définition 2.2.1.** *ils aiment les carottes*
- Définition 2.2.2** (des lapins). *ils aiment les carottes*

FIGURE 12.2 – Les théorèmes

Espace	Code
négative	<code>\!</code>
fine	<code>\,</code>
normale	<code>\ suivi d'une espace</code>
moyenne	<code>\;</code>
large	<code>\:</code>
cadratin	<code>\quad</code>
double cadratin	<code>\qquad</code>

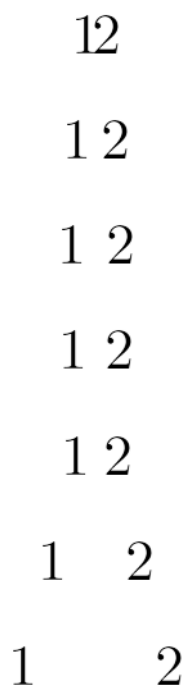


FIGURE 12.3 – Les espaces

Les accents et chapeaux divers

Les expressions scientifiques imposent parfois de coiffer une lettre ou une série de lettres de flèches (pour représenter des vecteurs) ou de divers accents ¹⁴. Voici un tableau les présentant.

Code	Rendu
<code>\hat{a}</code>	$\hat{a}$
<code>\acute{a}</code>	$\acute{a}$
<code>\bar{a}</code>	$\bar{a}$
<code>\dot{a}</code>	$\dot{a}$
<code>\breve{a}</code>	$\breve{a}$
<code>\check{a}</code>	$\check{a}$
<code>\grave{a}</code>	$\grave{a}$
<code>\vec{a}</code>	$\vec{a}$
<code>\ddot{a}</code>	$\ddot{a}$
<code>\tilde{a}</code>	$\tilde{a}$
<code>\overleftarrow{aze}</code>	$\overleftarrow{aze}$
<code>\overrightarrow{aze}</code>	$\overrightarrow{aze}$
<code>\overbrace{aze}</code>	$\overbrace{aze}$
<code>\underbrace{aze}</code>	$\underbrace{aze}$
<code>\overline{aze}</code>	$\overline{aze}$
<code>\underline{aze}</code>	$\underline{aze}$
<code>\stackrel{dessus}{dessous}</code>	$\stackrel{dessus}{dessous}$
<code>\widehat{aze}</code>	$\widehat{aze}$
<code>\widetilde{aze}</code>	$\widetilde{aze}$
<code>\xrightarrow[dessous]{dessus}</code>	$\xrightarrow[dessous]{dessus}$
<code>\xleftarrow[dessous]{dessus}</code>	$\xleftarrow[dessous]{dessus}$

En résumé

- Une expression mathématique peut être insérée au sein d'un paragraphe (il faut écrire `$expression$`) ou en dehors d'un paragraphe (il faut cette fois écrire `\[expression\]`).
- On utilise ensuite des commandes spécifiques pour mettre en forme les formules. Ainsi, `\frac{numérateur}{dénominateur}` permet de créer une fraction.
- Pour écrire un système d'équations, on utilise l'environnement `eqnarray` ou `align`.
- Il existe des centaines de caractères spéciaux dédiés à l'écriture d'expressions scientifiques. Reportez-vous à l'annexe A page 231 de ce livre pour en avoir la liste.

14. Dans le cas de `\stackrel`, nous remplacerons l'accent par une série de lettres.

Chapitre 13

Sommaire et index

Difficulté : >>>

Pre-
Prenons le temps de faire un bilan : les tableaux, les images, la mise en forme du texte et des pages... Nous serions tentés de nous dire que tous les grands domaines de $\text{\LaTeX}$ ont été abordés. Eh bien non, il nous en manque encore deux, qui nous occuperont pendant deux courts chapitres. Je pense ici aux différents types de tables d'index ainsi qu'aux bibliographies. Ces deux éléments, bien que ne représentant pas une difficulté insurmontable, sont le cauchemar de bon nombre d'étudiants et une intarissable source de blocage. Ne vous inquiétez pas, nous allons apprendre à utiliser simplement les tables et index dans ce chapitre¹. Préparez vos neurones !

Nous aborderons ici les tables de tableaux, de figures et bien sûr les tables des matières. Nous apprendrons aussi les mécanismes élémentaires nécessaires à la création d'un index simple.



1. Les bibliographies seront pour le chapitre suivant.

Tables des matières

Les tables sont censées aider le lecteur à mieux se repérer dans vos publications. Leur propreté (et accessoirement la vôtre) sera pour le correcteur ou le lecteur un gage de sérieux.

Dans de nombreux cas en édition, ce qui est jugé est votre table des matières. Mal la présenter revient à vous tirer, dès le départ, une balle dans le pied. C'est pour cela que tout projet sérieux requiert une table des matières réfléchie et bien fournie. Il est aussi parfois nécessaire d'enlever des éléments superflus à une table des matières, ou de ne pas aller trop loin dans le détail.

Imaginez un document où vous présentez à la fois vos chapitres, sections, sous-sections et paragraphes dans votre table des matières. Ne serait-ce pas un peu trop ? Nous apprendrons ici à enlever certains éléments de la table en demandant à L^AT_EX de ne pas y inclure les éléments hiérarchiques de trop bas niveau.

Résumons en trois mots les qualités d'une bonne table des matières :

- cohérence ;
- pertinence (vis-à-vis du contenu réel de la publication) ;
- propreté.

Table des matières simple

Avant de passer aux choses très subtiles, nous devons apprendre à inclure une table des matières dans un document. Pour ce faire, il vous suffit d'insérer la commande `\tableofcontents` à l'endroit où vous souhaitez afficher une table des matières dans votre document.

Je vous assure, ça n'a rien de plus compliqué. En une ligne de code, vous disposerez d'une table conçue avec les paramètres par défaut de L^AT_EX, ce qui suffit à la majorité des petits rapports. La seule subtilité réside dans le fait qu'il faille compiler deux fois. La première permet à L^AT_EX de comprendre la structure du document et de lister les titres. La seconde permet la mise en forme de la table, avec les numéros de pages.

Petite démonstration sur le code suivant (figure 13.1) :

```
\documentclass{book}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\begin{document}

\part{Partie 1}
  \chapter{Chapitre 1}
    \section*{Une section}
    \section{Une section avec un nom méga
      ↪ mais alors vraiment méga trop giga long qui dépasse}
    \subsection{Une sous-section}
```

```
\subsection{Une sous-section}
\chapter{Chapitre 2}
\section{Une section}
\subsection{Une sous-section}
\paragraph{Un paragraphe}
\subsection{Une sous-section}
\paragraph{Un paragraphe}
\section{Une section}
\chapter{Chapitre 3}
\section{Une section}
\section{Une section}

\tableofcontents

\end{document}
```

▷ Copier ce code
Code web : 772385

Table des matières

I	Partie 1	1
1	Chapitre 1	3
1.1	Une section avec un nom méga mais alors vraiment méga trop giga long qui dépasse	3
1.1.1	Une sous-section	3
1.1.2	Une sous-section	3
2	Chapitre 2	5
2.1	Une section	5
2.1.1	Une sous-section	5
2.1.2	Une sous-section	5
2.2	Une section	5
3	Chapitre 3	7
3.1	Une section	7
3.2	Une section	7

FIGURE 13.1 – Table des matières

La table obtenue apparaît après deux compilations et plusieurs éléments sont à noter :

- il manque les paragraphes ;
- le titre de l’une des sections est trop long ;

- il manque une section ;
- vous aimeriez appeler cette table des matières « Sommaire » et la placer en tête de document.

La seule de ces cinq choses que nous sachions faire pour l’instant est de mettre en tête de document la table des matières. Il suffit d’écrire `\tableofcontents` après la commande de création d’une éventuelle page de garde, ou encore juste après votre `\begin{document}`.

Examinons de plus près les autres éléments.

Paramétrage d’une table des matières

Appeler votre table « Sommaire »

Avec les packages francisant votre document, le nom par défaut de votre table des matières est « Table des matières ». Ce nom me semble peu adapté si la table est présente en début de manuscrit. L’appellation « Sommaire » peut paraître plus appropriée.

Voici la commande :

```
| \renewcommand{\contentsname}{Votre nouveau titre} % Dans le corps du document,  
| ↪ avant la commande \tableofcontents.
```

La figure 13.2 montre le résultat avec le mot « Sommaire ».

La méthode développée ici permet tout aussi bien de changer le nom en « Sommaire » qu’en autre chose. Elle est applicable aussi à peu près à tous les éléments de structure d’un document (pour changer l’appellation « Partie » en autre chose, il vous faudra remplacer `\contentsname` par `\partname` par exemple²).

Raccourcir une ligne

Nous pouvons voir que dans notre sommaire, une des lignes est trop longue et dépasse. Heureusement, la communauté L^AT_EX a pensé à permettre aux utilisateurs de définir deux titres différents lors de la création d’un élément de structure : l’un pour le document, l’autre pour la table des matières.

La commande `\section{titre dans le document}` se transforme en `\section[titre dans la table des matières]{titre dans le document}`.

Dans le cas qui nous intéresse, remplaçons

```
\section{Une section avec un nom méga mais alors vraiment méga trop giga  
long qui dépasse} par  
\section[un titre plus court]{ Une section avec un nom méga mais alors vrai-  
ment méga trop giga long qui dépasse} et observons le résultat sur la figure 13.3.
```

Magnifique, n’est-ce pas ? Et cela fonctionne avec tous les éléments de structure.

2. L’annexe « Mémento » rassemblera toutes ces commandes.

Sommaire

I	Partie 1	1
1	Chapitre 1	3
1.1	Une section avec un nom méga mais alors vraiment méga trop giga long qui dépasse	3
1.1.1	Une sous-section	3
1.1.2	Une sous-section	3
2	Chapitre 2	5
2.1	Une section	5
2.1.1	Une sous-section	5
2.1.2	Une sous-section	5
2.2	Une section	5
3	Chapitre 3	7
3.1	Une section	7
3.2	Une section	7

FIGURE 13.2 – Le changement de titre

Sommaire

I	Partie 1	1
1	Chapitre 1	3
1.1	Un titre plus court	3
1.1.1	Une sous-section	3
1.1.2	Une sous-section	3
2	Chapitre 2	5
2.1	Une section	5
2.1.1	Une sous-section	5
2.1.2	Une sous-section	5
2.2	Une section	5
3	Chapitre 3	7
3.1	Une section	7
3.2	Une section	7

FIGURE 13.3 – Les titres courts

Inclure les paragraphes dans le sommaire

Par défaut, $\text{\LaTeX}$ n'inclut pas les titres des paragraphes dans la table des matières de la classe `book`. Nous souhaiterions qu'ils apparaissent.

Pour ce faire, nous allons nous dire que chaque élément de hiérarchie correspond à un chiffre. Nous démarrons par les parties à -1 (je sais, c'est bizarre), puis les chapitres à 0 et ainsi de suite, cela nous donnant un tableau sympathique.

Élément hiérarchique	Nombre
Partie	-1
Chapitre	0
Section	1
Sous-section	2
Sous-sous-section	3
Paragraphe	4
Sous-paragraphe	5

La commande qui nous intéresse est `\setcounter{tocdepth}{Nombre de votre choix}`. Elle sert à demander à $\text{\LaTeX}$ de composer une table des matières contenant tous les éléments hiérarchiques jusqu'à ceux définis par le nombre que vous aurez choisi. Si vous choisissez -1 il n'y aura que les parties, si vous choisissez 5 il y aura tout. Tentons le 4 pour créer une table des matières comportant les titres des paragraphes :

```
\setcounter{tocdepth}{4}
```

Le résultat est visible sur la figure 13.4.

Inclure un élément de structure fictif

Lorsqu'un élément de structure est suivi de `*` (par exemple `\chapter*{un chapitre}`), il n'y a pas de référence vers celui-ci dans la table des matières. Néanmoins, il est aussi possible de forcer $\text{\LaTeX}$ à ajouter une entrée.

La commande est ³ `\addcontentsline{toc}{élément de structure sans son \ ni ses accolades et crochets}{titre apparaissant dans le sommaire}`.

Petit exemple (figure 13.5) :

```
\documentclass{book}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}
\begin{document}
\addcontentsline{toc}{part}{Le Monde}
```

3. Le `{toc}` est un élément que nous ne développerons pas, dites-vous juste que sans lui, la compilation peut échouer.

Sommaire

I	Partie 1	1
1	Chapitre 1	3
1.1	Un titre plus court	3
1.1.1	Une sous-section	3
1.1.2	Une sous-section	3
2	Chapitre 2	5
2.1	Une section	5
2.1.1	Une sous-section	5
	Un paragraphe	5
	Un paragraphe	5
	Un paragraphe	5
2.1.2	Une sous-section	5
	Un paragraphe	5
	Un paragraphe	5
	Un paragraphe	5
2.2	Une section	5
3	Chapitre 3	7
3.1	Une section	7
3.2	Une section	7

FIGURE 13.4 – Utilisation de `\setcounter`

```

\addcontentsline{toc}{chapter}{L'Eurasie}
\addcontentsline{toc}{section}{L'Europe}
\addcontentsline{toc}{subsection}{La France}
\addcontentsline{toc}{subsubsection}{L'Aquitaine}
\addcontentsline{toc}{paragraph}{La Gironde}
\addcontentsline{toc}{subparagraph}{Bordeaux}

\setcounter{tocdepth}{5}
\tableofcontents
\end{document}

```

▷ Copier ce code
Code web : 971974

Table des matières

Le Monde	1
L'Eurasie	1
L'Europe	1
La France	1
L'Aquitaine	1
La Gironde	1
Bordeaux	1

FIGURE 13.5 – Éléments de structure fictifs

Chose importante ici : étant donné que la commande `\addcontentsline` n'est qu'un marqueur et ne représente rien de réel dans le document, il est normal que le sommaire considère que tous les éléments sont en première page. Vous pouvez bien sûr les placer où vous le souhaitez dans votre document selon vos besoins.

Tables des figures et tableaux

Les tables des figures et des tableaux permettent au lecteur de mieux se retrouver dans un texte. Ces deux tables ne sont pas incontournables, mais suivent les mêmes mécanismes, à peu de chose près, que les tables des matières.

Elles s'insèrent respectivement via les commandes `\listoffigures` et `\listoftables` ; elles font apparaître, au choix, soit un titre spécialement conçu pour la table, soit la légende contenue dans `\caption`. Nous n'avons pas encore appris à indiquer des titres conçus pour les tables dans la commande `\caption`, pourtant c'est très simple. Les deux syntaxes possibles ci-dessous vous sont présentées.

```
\caption[titre court]{légende normale de l'image}
\caption{légende normale de l'image}
```

Vous vous attendiez à quelque chose de plus compliqué? Désolé de vous décevoir.



Ne vous trompez pas lors de la compilation. N'oubliez pas qu'il faut compiler deux fois. J'ai reçu de nombreux messages de personnes pensant être face à un bug de compilation dans leurs tables, alors qu'elles avaient simplement oublié la seconde compilation⁴.

Je vous ai fabriqué un exemple massif (figure 13.6), mais qui illustre bien la manière de créer des tables recensant les tableaux ainsi que les figures. Prenez le temps de le décortiquer, il n'est pas très difficile, mais très utile. J'ai réutilisé ici l'image de Poulpy, mascotte officielle du livre.

```
\documentclass[12pt]{report}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\usepackage{graphicx}
\begin{document}

\begin{table}
\begin{center}
\begin{tabular}{|c|c|}
\hline
1 & 2 \\
\hline
3 & 4 \\
\hline
\end{tabular}
\end{center}
\caption[Un tableau]{Mon beau tableau}
\end{table}

\begin{figure}
\begin{center}
\includegraphics{poulpy.png}
\end{center}
\caption{Poulpy est multicolore}
\end{figure}

\begin{figure}
\begin{center}
\includegraphics{poulpy.png}
```

4. Il y avait aussi quelques cas de mauvaise installation sur Windows.

```
\end{center}
\caption[Chatoyante]{Poulpy est chatoyante}
\end{figure}

\begin{figure}
\begin{center}
\includegraphics{poulpy.png}
\end{center}
\caption{Poulpy est inestimable}
\end{figure}

\begin{table}
\begin{center}
\begin{tabular}{|c|c|}
\hline
1 & 2 \\
\hline
3 & 4 \\
\hline
\end{tabular}
\end{center}
\caption{Mon beau tableau}
\end{table}

\begin{figure}
\begin{center}
\includegraphics{poulpy.png}
\end{center}
\caption[Poulpesque]{Poulpy est poulpesque}

\end{figure}

\listoftables
\listoffigures

\end{document}
```

▷ Copier ce code
Code web : 330464

Les index

À mon sens, les index constituent la limite en difficulté à aborder dans ce livre, et n'ont finalement pas un intérêt énorme dans des rapports ou de petites publications. J'ai tout de même choisi de vous en parler un peu, ne serait-ce que par souhait de proposer une ouverture.

Prenez conscience tout de même que je vous expose ici la méthode permettant d'élaborer des index élémentaires. Les choses peuvent vite devenir très difficiles à mettre en

Liste des tableaux

1	Un tableau	1
2	Mon beau tableau	1

Table des figures

1	Poulpy est multicolore	1
2	Chatoyante	2
3	Poulpy est inestimable	2
4	Poulpesque	3

FIGURE 13.6 – Table des figures et liste des tableaux

place.

En résumé, je vous explique ici le concept pour que vous ne soyez pas paniqué plus tard si vous devez approfondir la chose, mais nous ne nous y attarderons pas⁵.

Théorie

Commençons par les choses simples. La création d'un index nécessite le package `makeidx` et d'insérer la commande `\makeindex` dans le préambule, puis `\printindex` à l'endroit où vous souhaitez afficher celui-ci (exactement comme vous l'avez fait pour les tables).

Pour mettre en place les entrées dans l'index, il faut les marquer avec la commande `\index{argument}` en collant celle-ci au mot concerné. C'est à partir d'ici que les choses se gâtent.

| J'aime les patates`\index{patate}`.

Après trois compilations, vous vous retrouvez normalement avec un index indiquant la position du marqueur « patate » dans votre document.

5. Beaucoup de documentations n'abordent même pas le sujet.



Les utilisateurs de T_EXnicCenter n'auront besoin de compiler que trois fois pour créer un index. Néanmoins, ce n'est pas aussi simple pour tout le monde. Les utilisateurs de Linux devront compiler une fois avant de taper `makeindex nom_du_fichier.tex` dans leur console (ou cliquer sur Build → Compile → Makeindex dans Kile) et compiler de nouveau.

Les utilisateurs de T_EXShop trouveront quant à eux la commande « Makeindex » dans l'un de leurs menus déroulants. Ils devront eux aussi compiler normalement une fois, sélectionner « Makeindex » et compiler dans ce mode, puis compiler à nouveau en mode L^AT_EX.

Si nous insérons un marqueur « pâté » et un marqueur « patate », L^AT_EX placerait dans l'ordre alphabétique « pâté » avant « patate ». Difficile à croire, et pourtant (figure 13.7)...

| J'aime le pâté\index{pâté} et les patates\index{patate}.

Index

pâté, 1
patate, 1

FIGURE 13.7 – Index simple. Petit problème d'ordre alphabétique...

L'idée derrière tout ça c'est que L^AT_EX a un peu (voire beaucoup) de mal avec les accents. Il faut donc lui donner en argument une version non accentuée de chaque mot comportant des accents (c'est aussi valable pour les mots avec majuscules). Ces deux versions devront être séparées par un @. Oui, je sais : c'est tordu. Mais il faut faire avec, c'est le formalisme de L^AT_EX.

| J'aime le pâté\index{pate@pâté} et les patates\index{patate@Patate}.

La figure 13.8 montre le résultat du code vu plus haut. L'ordre alphabétique est de nouveau respecté.

Si vous souhaitez créer des sous-entrées dans l'index, il vous suffit de séparer les entrées et sous-entrées⁶ par des « ! ». L'exemple est très parlant ici (figure 13.9).

| J'aime le pâté\index{pate@pâté} et les patates\index{patate},
↪ ainsi que les patates nouvelles\index{patate!patates nouvelles}.

Index

Patate, 1
pâté, 1

FIGURE 13.8 – Tout s’arrange :)

Index

patate, 1
 patates nouvelles, 1
pâté, 1

FIGURE 13.9 – Sous-entrée dans un index

Pour finir, vous pouvez faire des références croisées au sein de vos index. « Références croisées », ce n'est pas vraiment très parlant comme terme, nous allons donc partir d'un exemple pour expliquer ce qu'elles sont. Faites le test, insérez `|see`. Cette commande permet à l'index de lier deux mots entre eux. Ainsi, écrire `\index{poulpy|see{bestiole mythologique}}` donnera dans l'index « poulpy, voir bestiole mythologique ».

Bien entendu, ici le mot « see » est traduit par « voir » lors de la compilation, mais L^AT_EX peut très bien compiler dans d'autres langues.

Pratique

Votre mission, si vous l'acceptez, est de créer l'index ci-dessous (figure 13.10) à partir de la phrase « Poulpy est chatoyante, merveilleuse et multicolore bien que son inestimable charisme ne cache qu'une fourberie sans nom, son être tout entier est abyssal et inestimable. »

Index

Poulpy, 1
 charisme, 1
 fourberie, 1
 chatoyante, 1
 être, 1
 abyssal, 1
 inestimable, 1
 merveilleuse, 1
 multicolore, 1

FIGURE 13.10 – Exercice

La solution

```
Poulpy\index{poulpy@Poulpy} est chatoyante\index{poulpy@Poulpy!chatoyante},  
↪ merveilleuse\index{poulpy@Poulpy!merveilleuse} et  
↪ multicolore\index{poulpy@Poulpy!multicolore} bien que son inestimable  
↪ charisme\index{poulpy@Poulpy!charisme} ne cache qu'une  
↪ fourberie\index{poulpy@Poulpy!charisme!fourberie} sans nom,  
↪ son être\index{poulpy@Poulpy!etre@être} tout entier
```

6. Notez que créer des sous-sous-entrées est possible.

```
↪ est abyssal\index{poulpy@Poulpy!etre@être!abyssal}
↪ et inestimable\index{poulpy@Poulpy!etre@être!inestimable}.
```

▷ Copier ce code
Code web : 970271

Si vous en êtes venu à bout sans problème, vous pouvez vous estimer à l’aise avec $\text{\LaTeX}$.

En résumé

- Une table des matières s’insère très facilement avec la commande `\tableofcontents`. De même, il est possible d’insérer la table des figures (`\listoffigures`) et des tables (`\listoftables`).
- Il faut compiler deux fois pour que ces tables puissent être générées.
- On utilise la commande `\index` pour ajouter des éléments à l’index. Pour compiler celui-ci, il faut appeler le programme `makeindex`.
- $\text{\LaTeX}$ place mal les mots accentués dans son index, mais il est possible d’y remédier en indiquant le même mot sans accent à $\text{\LaTeX}$.

Chapitre 14

La bibliographie

Difficulté : >>>

Cher lecteur, vous qui avez suivi avec attention jusque-là, je dois vous annoncer que vous arrivez au dernier chapitre de cours de ce livre.

Pour finir en beauté, nous allons apprendre l'une des notions qui donnent des boutons à bien des débutants¹ en $\text{\LaTeX}$: la bibliographie². Cet élément, bien que créé pour être maniable, est souvent mal compris par les novices et passé sous silence.

Conséquence logique du problème : les débutants n'insèrent pas de bibliographie dans leurs publications et ne citent ni leurs sources ni leurs ressources documentaires. C'est une grave erreur...



1. Les étudiants que j'ai interrogés m'ont pour la plupart avoué que ça leur faisait peur à tel point qu'ils n'avaient même pas essayé.

2. Ce chapitre est consacré entièrement à l'une des méthodes permettant de créer des bibliographies. Elle sera développée et expliquée en long, en large et en travers.

Principe d'une bibliographie

Avant d'aborder la technique et $\text{\LaTeX}$, j'aimerais parler de la définition du mot « bibliographie » et de son sens. Rien que sur la signification du mot, deux écoles se distinguent.

D'un côté, nous avons ceux qui estiment que la bibliographie comporte la liste de toutes les œuvres, qu'elles soient sous format papier ou numérique, qui ont servi de près ou de loin à concevoir une production. De l'autre, il y a ceux pour qui une bibliographie est un rassemblement d'œuvres répondant à un besoin particulier. Je vous cite ci-dessous un extrait de la définition du mot « bibliographie » sur Wikipédia, qui vous permettra de vous faire une idée des différents types de bibliographie existants.

Une bibliographie thématique regroupe des documents sur un même sujet ou un même thème. Inversement, une bibliographie systématique (parfois générale) regroupe des documents parus dans un même lieu et / ou à une même date.

Une bibliographie signalétique ne donne que les caractéristiques essentielles du document. Une bibliographie analytique en donne une description qui se veut neutre. Une bibliographie critique émet un jugement sur les documents qui y sont signalés.

Une bibliographie rétrospective décrit des documents publiés dans le passé.

Une bibliographie exhaustive décrit tous les livres publiés dans le cadre thématique, chronologique ou géographique retenu ; elle s'oppose à une bibliographie sélective. Une bibliographie sommaire propose une description plus réduite des ouvrages concernés.

La forme globale et la présentation d'une bibliographie sont **variables selon les besoins et les auteurs de contenu**. C'est ce que nous devons retenir ici.

À ce stade, normalement, les choses doivent vous sembler un peu tordues. Vous pensiez savoir ce qu'est une bibliographie. . . et au final vous vous rendez compte que les bibliographies peuvent prendre de multiples formes. Néanmoins, ces multiples formes ont un point commun : **la mise en forme d'une liste d'œuvres**.

Restons sur cette idée-là : « Une bibliographie est une liste d'un certain nombre d'œuvres choisies, répondant toutes à une consigne (regrouper les sources, des œuvres thématiques, etc.), mise en page selon les souhaits d'un (ou de plusieurs) auteur(s) ».

Tout de suite, nous y voyons déjà un peu plus clair.

La démarche de l'auteur est donc de choisir, dans la longue liste des œuvres qu'il connaît, celles qui seront publiées dans sa bibliographie. La mise en forme, $\text{\LaTeX}$ sait très bien la faire. La liste proprement dite est créée par un programme externe : $\text{\BibTeX}$.



$\text{\BibTeX}$ fournit à l'auteur une armada d'outils destinés à concevoir facilement sa liste (que nous appellerons base de données) contenant toutes les informations sur les œuvres dont il dispose (auteur, date de parution, etc.).

L'auteur affectera une petite étiquette à chaque œuvre. Il lui suffira ensuite, dans son document $\text{\LaTeX}$, de faire référence à certaines étiquettes pour intégrer aussitôt les œuvres concernées dans la bibliographie de sa publication.

Résumons tout ceci dans un schéma (figure 14.1).

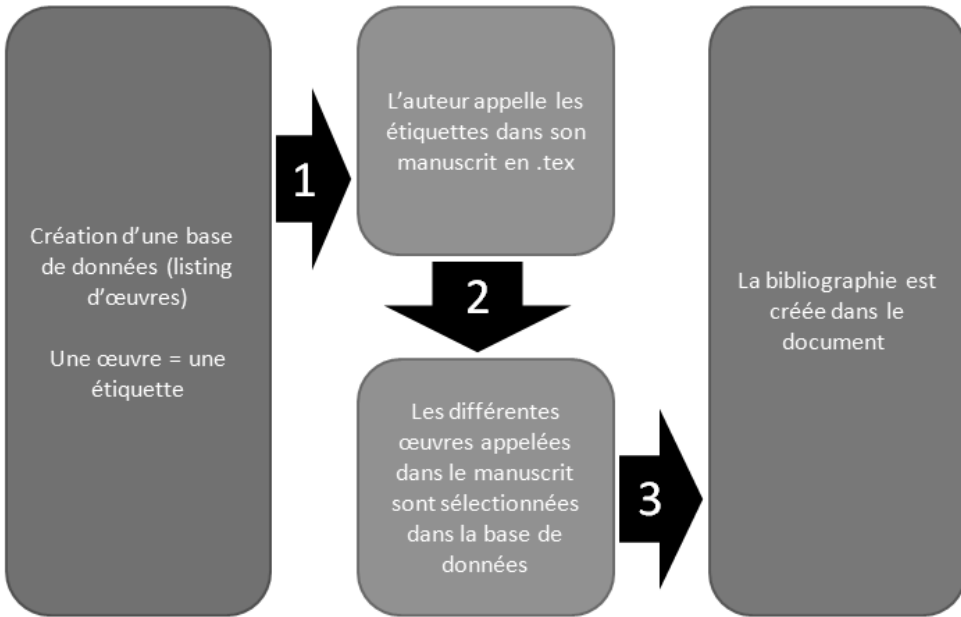


FIGURE 14.1 – Schéma résumé

L'idée générale étant maintenant fixée, entrons dans le vif du sujet. Dans un premier temps, nous créerons la base de données, avant d'apprendre à faire des références à celle-ci dans nos documents `.tex` pour créer la bibliographie.

La base de données

Créer la base de données est la partie la plus facile. Elle est constituée de paragraphes contenant chacun les informations et l'étiquette des œuvres.



Le fichier que nous enregistrerons ici, qui contiendra la base de données, ne sera pas un `.tex` mais un `.bib`.

La structure des informations concernant une œuvre dans la base de données est toujours la même. Je vous la montre ci-dessous.

```
|@type_d'œuvre{référence_courte,
```

```
Titre_du_champ1={texte de votre choix},
Titre_du_champ2={texte de votre choix},
Titre_du_champ3={texte de votre choix},
Titre_du_champ4={texte de votre choix},
}
Entre deux œuvres, mon texte est considéré comme commentaire.
@type_d'œuvre{référence_courte,
Titre_du_champ1={texte de votre choix},
Titre_du_champ2={texte de votre choix},
Titre_du_champ3={texte de votre choix},
Titre_du_champ4={texte de votre choix},
}
```

Vous pouvez enchaîner autant d'œuvres que vous le souhaitez dans le fichier. Il faut néanmoins respecter certaines règles pour que votre base de données fonctionne correctement :

- ne pas oublier les virgules ;
- ne pas choisir au hasard des types d'œuvres ou des titres de champs. Ils sont prédéfinis, vous ne pouvez pas en inventer.

Voilà ce qui rebute les débutants : le formalisme un peu triste et strict. En réalité, par rapport à la mise en forme d'un tableau, il est beaucoup plus simple d'écrire et de relire ce genre de code.

Revenons à notre code. Je vous ai dit que les types d'œuvres³ et les titres des champs étaient prédéfinis. Il est temps pour vous de jeter un œil à leurs deux tableaux récapitulatifs.

Types d'œuvres		
Type d'œuvre	Correspondance	Champs disponibles
@article	Article	author, journal, title, year, month, pages, notes
@book	Livre	author, title, publisher, year
@manual	Document technique	title, author, year, organization
@misc	Divers	author, title, month, year, note
@phdthesis	Thèse de doctorat	title, author, school, year
@unpublished	Manuscrit non publié	author, title, note, year, month

Il en existe bien sûr d'autres, mais ceux-ci sont les plus répandus.

Nous pouvons, à l'aide des deux tableaux, créer notre première petite base de données bibliographique. Nous l'appellerons `biblio.bib`.

```
Pour author, les noms et prénoms sont séparés par une vigule, et les différents
↪ auteurs par AND
```

3. Le type @misc est utilisé pour les sites web et ce qui ne trouve pas sa place dans les autres catégories.

Champs	
Champs	Correspondance
author	Auteur
journal	Journal
month	Mois de publication, au choix (jan, feb, mar, apr...)
note	Notes complémentaires
organization	Société ou organisation
pages	Pages concernées
publisher	Éditeur
year	Année de publication

```
@Article{ref,  
  author={Durand, Patrick AND Durand, René},  
  journal={Le beau journal},  
  title={Les tomates tueuses},  
  year={2007},  
  month={jan},  
  pages={24},  
  notes={sympathique}  
}
```

```
@book  
{ref2,  
  author={Dupont, Fernand},  
  title={Les choux farcis},  
  publisher={Un gros éditeur},  
  year={2004}  
}
```

```
@manual  
{ref3,  
  title={Réparer son vaisseau},  
  author={Dupont, Nestor},  
  year={2009},  
  organization={L'Alliance}  
}
```

```
@misc  
{ref4,  
  author={Les Zéros},  
  title={Le site du zéro},  
  month={jun},  
  year={2009},  
  note={www.siteduzero.com}  
}
```

```
@phdthesis  
{ref5,
```

```
title={La belle thèse},
author={Barbu, Monsieur},
school={Université},
year={2002}
}

@unpublished
{ref6,
author={Mauvais, Monsieur},
title={Mon roman inachevé},
note={il est chouette mon roman},
year={2000},
month={feb}
}
```

▷ Copier ce code
Code web : 268036

Mise en place de la bibliographie

La base de données bibliographique que vous venez d'apprendre à créer peut contenir autant d'éléments que vous le souhaitez. L'idée centrale est d'y regrouper les œuvres sur lesquelles vous vous basez et celles que vous citerez dans vos documents.

Ici nous avons créé un fichier `.bib`. Certains n'en utilisent qu'un seul dans lequel ils vont piocher ⁴, d'autres en créent un par publication, c'est selon les goûts de chacun. De mon côté, j'aime assez l'idée de disposer, en un seul endroit, de la liste des ressources utilisées dans tous mes écrits ⁵.

Concept

Apprenons maintenant à nous servir de cette base. Nous ferons des références à des éléments de la base de données via la commande `\cite{référence_courte}` que nous placerons où nous le voudrions dans le texte. Deux autres commandes viennent s'y ajouter : `\bibliographystyle{ }` et `\bibliography{nom du fichier.bib sans l'extension .bib}`. La première donne à $\text{\LaTeX}$ le style de mise en page souhaité pour la bibliographie, la seconde lui précise l'emplacement de la base de données.

Les éléments de la base de données auxquels les commandes `\cite{référence_courte}` font référence seront listés avec leurs informations respectives dans une page « Bibliographie » ou un encart « Références », suivant les classes de vos documents. Cette page se trouvera au même emplacement que celui où vous choisirez de faire apparaître les

4. L'intérêt d'une base de données est justement de pouvoir piocher à l'intérieur sans être obligé d'utiliser l'intégralité des informations qu'elle stocke.

5. En partant du principe que le geek a tendance à être un peu désordonné, disséminer des fichiers `.bib` partout est dangereux.

commandes `\bibliographystyle{ }` et `\bibliography{ }`⁶.

Exemples

Beaucoup d'informations vous ont été données. Il est temps de vous montrer quelques exemples, illustrant les différents styles de bibliographies possibles, avant de vous expliquer plus en détail comment se passe la compilation sous Mac OS, Linux ou Windows.

```
\documentclass{article}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\begin{document}

J'insère le premier \cite{ref}, le second \cite{ref2}, le troisième \cite{ref3},
↔ le quatrième \cite{ref4}, le cinquième \cite{ref5} et le sixième \cite{ref6}.

\bibliographystyle{} % Le style est mis entre accolades.
\bibliography{bibli} % mon fichier de base de données s'appelle bibli.bib

\end{document}
```

▷ Copier ce code
Code web : 663237

`\bibliographystyle{plain}` classe les entrées par ordre alphabétique et les numérote en conséquence (figure 14.2).

`\bibliographystyle{abbrv}` classe les entrées par ordre alphabétique, les numérote en conséquence et abrège certains éléments de la bibliographie (figure 14.3).

`\bibliographystyle{unsrt}` trie les entrées par ordre d'apparition dans le texte (figure 14.4).

`\bibliographystyle{alpha}` : le repère n'est plus un chiffre, mais les trois premières lettres du nom de l'auteur accolées aux deux derniers chiffres de l'année de parution (figure 14.5).

Compilation

La compilation d'un document comportant une bibliographie se passe d'une manière légèrement inhabituelle. Détaillons les trois façons de venir à bout de celle-ci.

Gardons à l'esprit qu'il faudra d'abord compiler notre document $\text{\LaTeX}$ (afin d'obtenir la liste des références à rechercher dans la base de données) avant d'appeler BibTeX

6. Leur fonctionnement est similaire à celui des tables.

J'insère le premier [4], le second [2], le troisième [3], le quatrième [6], le cinquième [1] et le sixième [5].

Références

- [1] Monsieur Barbu. *La Belle thèse*. PhD thesis, Université, 2002.
- [2] Fernand Dupont. *Les choux farcis*. Un gros éditeur, 2004.
- [3] Nestor Dupont. *réparer son vaisseau*. l'Alliance, 2009.
- [4] Patrick Durand and René Durand. Les tomates tueuses. *Le beau journal*, page 24, jan 2007.
- [5] Monsieur Mauvais. Mon roman inachevé. il est chouette mon roman, feb 2000.
- [6] Les Zéros. Le site du zéro, jun 2009. www.siteduzero.com.

FIGURE 14.2 – `\bibliographystyle{plain}`

J'insère le premier [4], le second [2], le troisième [3], le quatrième [6], le cinquième [1] et le sixième [5].

Références

- [1] M. Barbu. *La Belle thèse*. PhD thesis, Université, 2002.
- [2] F. Dupont. *Les choux farcis*. Un gros éditeur, 2004.
- [3] N. Dupont. *réparer son vaisseau*. l'Alliance, 2009.
- [4] P. Durand and R. Durand. Les tomates tueuses. *Le beau journal*, page 24, jan 2007.
- [5] M. Mauvais. Mon roman inachevé. il est chouette mon roman, feb 2000.
- [6] L. Zéros. Le site du zéro, jun 2009. www.siteduzero.com.

FIGURE 14.3 – `\bibliographystyle{abbrv}`

J'insère le premier [1], le second [2], le troisième [3], le quatrième [4], le cinquième [5] et le sixième [6].

Références

- [1] Patrick Durand and René Durand. Les tomates tueuses. *Le beau journal*, page 24, jan 2007.
- [2] Fernand Dupont. *Les choux farcis*. Un gros éditeur, 2004.
- [3] Nestor Dupont. *réparer son vaisseau*. l'Alliance, 2009.
- [4] Les Zéros. Le site du zéro, jun 2009. www.siteduzero.com.
- [5] Monsieur Barbu. *La Belle thèse*. PhD thesis, Université, 2002.
- [6] Monsieur Mauvais. Mon roman inachevé. il est chouette mon roman, feb 2000.

FIGURE 14.4 – `\bibliographystyle{unsrt}`

J'insère le premier [DD07], le second [Dup04], le troisième [Dup09], le quatrième [Zér09], le cinquième [Bar02] et le sixième [Mau00].

Références

- [Bar02] Monsieur Barbu. *La Belle thèse*. PhD thesis, Université, 2002.
- [DD07] Patrick Durand and René Durand. Les tomates tueuses. *Le beau journal*, page 24, jan 2007.
- [Dup04] Fernand Dupont. *Les choux farcis*. Un gros éditeur, 2004.
- [Dup09] Nestor Dupont. *réparer son vaisseau*. l'Alliance, 2009.
- [Mau00] Monsieur Mauvais. Mon roman inachevé. il est chouette mon roman, feb 2000.
- [Zér09] Les Zéros. Le site du zéro, jun 2009. www.siteduzero.com.

FIGURE 14.5 – `\bibliographystyle{alpha}`

(pour piocher dans la base de données) et de compiler une nouvelle fois notre document $\text{\LaTeX}$.

Dans la pratique, des cas exotiques peuvent survenir⁷ : j’ai pris l’habitude de compiler deux fois au lieu d’une les sources $\text{\LaTeX}$.

Sous $\text{\MiKTeX}$

Compilez deux fois votre document, comme vous le feriez d’habitude, avant de cliquer sur le bouton « $\text{\BibTeX}$ » contenu dans **Build** → **Current File** (figure 14.6) et de compiler deux nouvelles fois.

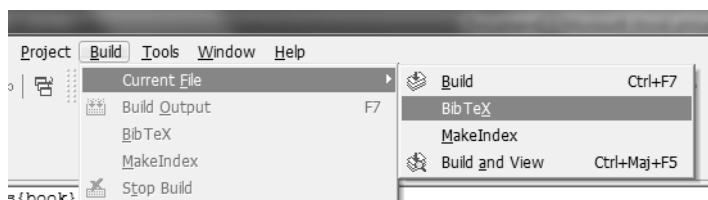


FIGURE 14.6 – Le bouton bibtex de $\text{\MiKTeX}$

Le processus est un peu fastidieux, mais une fois maîtrisé il ne pose pas de problème : il s’agit juste d’une nouvelle routine à intégrer.

Sous Linux

Pour vous, cela se passe dans la console avec les commandes suivantes :

```
latex nom_de_fichier.tex
latex nom_de_fichier.tex

bibtex nom_de_fichier

latex nom_de_fichier.tex
latex nom_de_fichier.tex
```

$\text{\TeX}$ Shop

Sous Mac OS, compilez deux fois comme à votre habitude dans un premier temps. Sélectionnez ensuite, dans le menu déroulant approprié (voir figure 14.7), l’option « $\text{\BibTeX}$ » au lieu de « $\text{\LaTeX}$ » et compilez. Cela fait, sélectionnez à nouveau $\text{\LaTeX}$ dans votre menu déroulant, compilez deux nouvelles fois et tout ira bien.

7. Typiquement, des créations de sommaires, de notes de bas de page et autres petites choses sympathiques.

FIGURE 14.7 – Le bouton BibTeX de T_EXShop

En résumé

- Créer une bibliographie nécessite une nouvelle composante de notre installation L^AT_EX : BibT_EX.
- Une bibliographie se construit en deux temps : d’abord, il faut élaborer une base de données en créant un fichier `.bib`. Ensuite, on fait appel aux références dans le texte avec la commande `\cite{}`.
- La bibliographie est insérée avec la commande `\bibliography{}`.
- Il existe quatre façons différentes de présenter une bibliographie. La commande `\bibliographystyle{}` permet de choisir entre ces différentes options de mise en forme.
- Une bibliographie **n’est pas un élément facultatif** dans un rapport ou une thèse, votre crédibilité peut en dépendre. Ne la négligez pas!

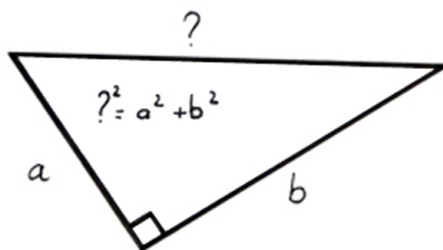
Chapitre 15

TP 2 – Rédaction d'un article présentant le théorème de Pythagore

Difficulté : 

Vous voici arrivé au deuxième et dernier TP de cet ouvrage. Il a été pensé pour qu'une majeure partie des notions abordées précédemment soit revue et que vous puissiez vous rendre compte de vos lacunes éventuelles. Nous utiliserons ici un article de Wikipédia traitant du théorème de Pythagore et deux tables issues du même site.

Le livre final issu de ce TP fait 23 pages (dont 9 blanches) et ne comporte pas de piège vicieux. J'ai préféré vous faire utiliser les commandes usuelles, à vous de voir si vous voulez ajouter des fioritures ou faire des essais¹.



1. Je vous conseille d'être curieux et de tenter des choses.

Les consignes

J'ai choisi pour cet exercice de ne plus vous tenir la main, il vous faudra réussir à suivre les consignes données, tenter de recréer une structure et la remplir avec le contenu indiqué, ceci sans oublier de respecter les standards et le formalisme de L^AT_EX.

Terrifiant, n'est-ce pas ?

Ne vous inquiétez pas, tout va bien se passer.

La structure

Nous allons nous préoccuper en premier lieu de la structure du **livre** que nous rédigeons. Il comportera :

- une page de garde dont le titre sera « Le théorème de Pythagore » et comportant votre nom en **petites capitales** et la **date du jour** ;
- un **sommaire** ;
- un **chapitre d'introduction** que l'on appellera « Introduction » ;
- une **première partie** nommée « Théorème de Pythagore » contenant deux **chapitres**, l'un nommé « Énoncé du théorème » et l'autre « Réciproque » ;
- le **chapitre** « Énoncé du théorème » comportera deux **sections** appelées respectivement « Théorie » et « Exemple » ;
- une **seconde partie**, du nom d'« Annexes et tables » contiendra : un **chapitre** « Table d'addition », un **chapitre** « Table de multiplication », une **table des figures**, une **liste des tableaux** ainsi qu'une **bibliographie**.

Vous devrez choisir où placer les différentes commandes de structure relatives aux livres selon vos souhaits. Essayez de rester le plus pertinent possible dans vos choix.

Ci-dessous le code web pointant vers l'article « Théorème de Pythagore » sur Wikipédia (duquel est issu le texte que nous utiliserons).

▷

Théorème de Pythagore
Code web : 941605

Les contenus

Chapitre d'introduction

Le chapitre d'introduction comporte un paragraphe et une image (dont la légende sera « Une version géométrique du théorème »).

Le texte qu'il doit contenir est le suivant et l'image (figure 15.1) est placée sur la page de l'article.

Le théorème de Pythagore est un théorème de géométrie euclidienne qui énonce que dans un triangle rectangle (qui possède un angle droit) le carré de la longueur de l'hypoténuse (côté opposé à l'angle droit) est égal à la somme des carrés des longueurs des deux autres côtés. Ce théorème est

nommé d'après Pythagore de Samos, mathématicien, philosophe et astronome de la Grèce antique.

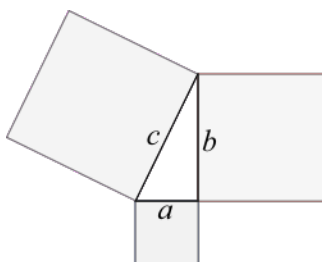


FIGURE 15.1 – Image de l'introduction

Chapitre « Énoncé du théorème »

Nous utiliserons ici pour la première fois dans notre document une référence à la bibliographie, ainsi que la mise en page liée au théorème (figure 15.2). Les mathématiques sont omniprésentes et un flottant fait son apparition (figure 15.3). Bien que déclaré après les paragraphes de texte, ce flottant est inséré dans la section « Théorie ».

Le texte provient de l'article « Théorème de Pythagore » de Wikipédia et la mise en page doit respecter les captures ci-dessous (eh non, je ne vous donne pas l'image, vous allez devoir la prendre sur la page du wiki et la convertir vous-même en PNG).

Chapitre « Réciproque »

Si vous avez réussi l'épreuve du chapitre « Énoncé du théorème », celui-ci ne vous posera aucun problème (figure 15.4).

Tables d'addition et de multiplication

Ces deux annexes (figures 15.5 et 15.6) nécessitent la création de tableaux, l'utilisation d'une couleur en niveaux de gris et des références bibliographiques. Les choses se corsent un peu. . .

La bibliographie

Une bibliographie sera utile dans ce document, je vous montre son allure sur la figure 15.7. À vous de créer la base de données correspondante².

Au boulot (normalement, vous ne devriez pas avoir besoin de plus d'une bonne heure) !

2. Elle ne comporte aucun piège.

Chapitre 1

Énoncé du théorème

1.1 Théorie

La forme la plus connue du théorème de Pythagore [3] est la suivante :

Théorème (de Pythagore). *Dans un triangle rectangle, le carré de la longueur de l'hypoténuse (côté opposé à l'angle droit) est égal à la somme des carrés des longueurs des côtés de l'angle droit.*

Note : Le terme « longueur », généralement oublié, est très important. En effet, la longueur est un nombre réel sur lequel l'opération d'élevation au carré est parfaitement définie; l'hypoténuse est un segment de droite, objet géométrique pour lequel l'élevation au carré n'a pas de sens.

Cependant, il est parfois retiré afin de ne pas compliquer l'apprentissage du théorème (La notion de longueur étant sous-entendue).

Dans un triangle ABC rectangle en C, AB étant l'hypoténuse, où $AB = c$, $AC = b$ et $BC = a$ (cf. figure ci-dessus), on aura donc : $BC^2 + AC^2 = AB^2$ ou encore $a^2 + b^2 = c^2$.

Le théorème de Pythagore permet ainsi de calculer la longueur d'un des côtés d'un triangle rectangle si on connaît les deux autres.

1.2 Exemple

Avec les notations ci-dessus, soit le triangle rectangle de côtés $a = 3$ et $b = 4$; alors la longueur du troisième côté, c , est donnée par : $a^2 + b^2 = 3^2 + 4^2 = 25 = c^2$. Les longueurs étant des réels positifs, on obtient $c = 5$. Un triplet de nombres entiers tel que $(3, 4, 5)$, représentant la longueur des côtés d'un triangle rectangle s'appelle un triplet pythagoricien.

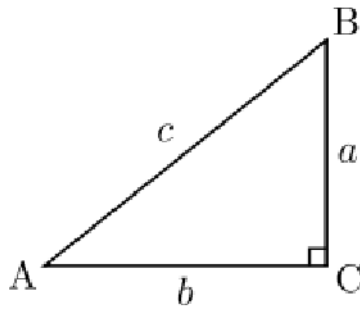


FIGURE 1.1 – Triangle rectangle

FIGURE 15.3 – Le flottant

Chapitre 2

Réciproque

La réciproque du théorème de Pythagore est également vraie :

Réciproque (Théorème de Pythagore). *Si dans un triangle, la somme des carrés de deux côtés est égale au carré du plus grand côté, alors ce triangle est rectangle.*

Le théorème de Pythagore est donc une propriété caractéristique des triangles rectangles. Formulé autrement, si dans un triangle ABC on a $BC^2 + AC^2 = AB^2$, alors ce triangle est rectangle en C .

Annexe A

Table d'addition

Table issue de Wikipédia [1].

Additionné à	1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10	11
2	3	4	5	6	7	8	9	10	11	12
3	4	5	6	7	8	9	10	11	12	13
4	5	6	7	8	9	10	11	12	13	14
5	6	7	8	9	10	11	12	13	14	15
6	7	8	9	10	11	12	13	14	15	16
7	8	9	10	11	12	13	14	15	16	17
8	9	10	11	12	13	14	15	16	17	18
9	10	11	12	13	14	15	16	17	18	19
10	11	12	13	14	15	16	17	18	19	20

TABLE A.1 – Table d'addition

FIGURE 15.5 – Table d'addition

Annexe B

Table de multiplication

Table issue de Wikipédia [2].

Multiplié par	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

TABLE B.1 – Table de multiplication

11

FIGURE 15.6 – Table de multiplication

Bibliographie

- [1] Wikipédia. Table d'addition, jul 2010. fr.wikipedia.org.
- [2] Wikipédia. Table de multiplication, jul 2010. fr.wikipedia.org.
- [3] Wikipédia. Théorème de pythagore, jul 2010. fr.wikipedia.org.

FIGURE 15.7 – La bibliographie

Petite aide

Vous avez eu toutes les informations nécessaires à la création du document, néanmoins, je me doute qu'il n'est pas forcément évident de visualiser le travail dans sa globalité. Si (et seulement si) vous avez essayé de faire l'exercice, mais que vous vous sentez trop perdu, vous pouvez consulter le document au format PDF fourni via le code web ci-dessous.

▷

Voir le résultat (PDF)
 Code web : 611171

La correction

Compiler souvent, reprendre les explications des chapitres précédents et les exemples, telles étaient les clés du succès. Je ne vous donnerai pas ici de technique détaillée pour parvenir au document final³.

Les concepts avancés de L^AT_EX s'entremêlent tous lors de la composition d'un document, de sorte que vous pouvez indifféremment constituer votre bibliographie pendant ou après l'écriture de votre contenu. L'important, c'est d'être organisé et de ne rien oublier. Lors de la création du TP, j'ai utilisé la méthode suivante pour réaliser le document :

- création de la structure d'un livre (avec `\appendix`, `\begin{document}`, etc.);
- compilation et correction des erreurs;
- page de garde et listing des `\usepackage{}`;
- compilation et correction des erreurs;
- création des éléments de structure interne (parties, chapitres, sections);
- compilation et correction des erreurs;

3. C'est exactement le même type de processus que pour le premier TP.

- création du sommaire ;
- double compilation et correction des erreurs ;
- remplissage des différents chapitres ;
- compilation et correction des (nombreuses) erreurs ;
- création des flottants ;
- compilation et correction des erreurs ;
- création des tables de flottants ;
- compilation ;
- création de la bibliographie ;
- cycle habituel : deux compilations, appel de BibT_EX puis deux nouvelles compilations.

Le trait est ici à peine grossi, et la méthode est ponctuée par de nombreuses étapes de correction. Cette méthode a l’avantage de pouvoir être transposée à la quasi-totalité des ébauches de publications que vous rédigerez ⁴. Suivant votre efficacité, vous pourrez sauter l’une ou l’autre des compilations.

Pour ma part, j’aime assez savoir d’où viennent mes erreurs et compiler après chaque étape me permet de ne pas avoir besoin de fouiller très longtemps, même avec des documents complexes.

Les deux seules difficultés de cet exercice sont la présence d’une bibliographie à remplir avec des pages web (donc, des entrées de type `@misc`) et les tableaux un peu complexes (qui ont déjà été vus dans le chapitre sur les tableaux). Je vous livre ci-dessous les deux codes source, celui de `bibliographie.bib`, ainsi que celui de `tp2.tex`. Prenez le temps de les lire si vous avez bloqué dessus et réessayez : vous y arriverez.

```
\documentclass{book}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
\usepackage{graphicx}
\usepackage{amsthm}
\usepackage{amsmath}
\usepackage{amssymb}
\usepackage{mathrsfs}
\usepackage{color}
\usepackage{colortbl}

\definecolor{grisclair}{gray}{0.8}

\newtheorem*{theo}{Théorème}
\newtheorem*{reci}{Réciproque}

\title{Le théorème de Pythagore}
\author{\textsc{Laleloulilo}}
\date{\today}
```

4. Évidemment, des étapes peuvent venir se greffer ou ne pas s’avérer nécessaires.

```

\begin{document}

\renewcommand{\contentsname}{Sommaire}

\maketitle % Page de garde.

\frontmatter
\tableofcontents

\chapter{Introduction}

Le théorème de Pythagore est un théorème de géométrie euclidienne qui énonce
↪ que dans un triangle rectangle (qui possède un angle droit) le carré de la
↪ longueur de l'hypoténuse (côté opposé à l'angle droit) est égal à la somme
↪ des carrés des longueurs des deux autres côtés. Ce théorème est nommé
↪ d'après Pythagore de Samos, mathématicien, philosophe et astronome de la
↪ Grèce antique.

\begin{figure}
\begin{center}
\includegraphics{intro.png}
\end{center}
\caption{Une version géométrique du théorème}
\label{Une version géométrique du théorème}
\end{figure}

\mainmatter
\part{Théorème de Pythagore}
\chapter{Énoncé du théorème}

\section{Théorie}

La forme la plus connue du théorème de Pythagore \cite{theo} est la suivante :

\begin{theo}[de Pythagore]
Dans un triangle rectangle, le carré de la longueur de l'hypoténuse (côté
↪ opposé à l'angle droit) est égal à la somme des carrés des longueurs
↪ des côtés de l'angle droit.
\end{theo}

Note : Le terme « longueur », généralement oublié, est très important.
↪ En effet, la longueur est un nombre réel sur lequel l'opération
↪ d'élévation au carré est parfaitement définie ; l'hypoténuse est
↪ un segment de droite, objet géométrique pour lequel l'élévation
↪ au carré n'a pas de sens.

Cependant, il est parfois retiré afin de ne pas compliquer l'apprentissage du
↪ théorème (la notion de longueur étant sous-entendue).

```

```
\begin{figure}
\begin{center}
\includegraphics{triangle.png}
\end{center}
```

```
\caption{Triangle rectangle}
\label{Triangle rectangle}
\end{figure}
```

Dans un triangle ABC rectangle en C, AB étant l’hypoténuse, où $AB = c$,
 $\hookrightarrow AC = b$ et $BC = a$ (cf. figure ci-dessus), on aura donc :
 $\hookrightarrow BC^2 + AC^2 = AB^2$ ou encore $a^2 + b^2 = c^2$.

Le théorème de Pythagore permet ainsi de calculer la longueur d’un des côtés
 $\hookrightarrow$ d’un triangle rectangle si on connaît les deux autres.

```
\section{Exemple}
```

Avec les notations ci-dessus, soit le triangle rectangle de côtés $a = 3$
 $\hookrightarrow$ et $b = 4$; alors la longueur du troisième côté, c , est donnée par :
 $\hookrightarrow a^2 + b^2 = 3^2 + 4^2 = 25 = c^2$. Les longueurs étant des réels positifs,
 $\hookrightarrow$ on obtient $c = 5$. Un triplet de nombres entiers tel que $(3, 4, 5)$,
 $\hookrightarrow$ représentant la longueur des côtés d’un triangle rectangle s’appelle
 $\hookrightarrow$ un triplet pythagoricien.

```
\chapter{Réciproque}
```

La réciproque du théorème de Pythagore est également vraie :

```
\begin{reci}[Théorème de Pythagore]
Si dans un triangle, la somme des carrés de deux côtés est égale au carré du
 $\hookrightarrow$  plus grand côté, alors ce triangle est rectangle.
\end{reci}
```

Le théorème de Pythagore est donc une propriété caractéristique des triangles
 $\hookrightarrow$ rectangles. Formulé autrement, si dans un triangle ABC on a
 $\hookrightarrow BC^2 + AC^2 = AB^2$, alors ce triangle est rectangle en C .

```
\appendix
\part{Annexes et Tables}
```

```
\chapter{Table d’addition}
```

Table issue de Wikipédia \cite{addi}.

```
\begin{table}
\begin{center}
```

```
\begin{tabular}{>{\begin{bf} \columncolor{grisclair}} c <{\end{bf}}ccccccccc}
```

```

\rowcolor{grisclair}Additionné à & \begin{bf}1\end{bf} & \begin{bf}2\end{bf}
↪ & \begin{bf}3\end{bf} & \begin{bf}4\end{bf} & \begin{bf}5\end{bf}
↪ & \begin{bf}6\end{bf} & \begin{bf}7\end{bf} & \begin{bf}8\end{bf}
↪ & \begin{bf}9\end{bf} & \begin{bf}10\end{bf} \\

1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\
4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 \\
5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\
6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 \\
8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 \\
9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 \\
10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\

\end{tabular}
\end{center}
\caption{Table d'addition}
\label{Table d'addition}
\end{table}

\chapter{Table de multiplication}

Table issue de Wikipédia \cite{mult}.

\begin{table}
\begin{center}

\begin{tabular}{>{\begin{bf} \columncolor{grisclair}} c <{\end{bf}}ccccccccc}

\rowcolor{grisclair}Multiplié par & \begin{bf}1\end{bf} & \begin{bf}2\end{bf}
↪ & \begin{bf}3\end{bf} & \begin{bf}4\end{bf} & \begin{bf}5\end{bf}
↪ & \begin{bf}6\end{bf} & \begin{bf}7\end{bf} & \begin{bf}8\end{bf}
↪ & \begin{bf}9\end{bf} & \begin{bf}10\end{bf} \\

1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
2 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 & 20

```

```

3 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 & 30 \\
4 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 & 40 \\
5 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 \\
6 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 & 60 \\
7 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 & 70 \\
8 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 & 80 \\
9 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 & 90 \\
10 & 10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 \\

\end{tabular}
\end{center}
\caption{Table de multiplication}
\label{Table de multiplication}
\end{table}
\backmatter

\listoffigures
\listoftables

\bibliographystyle{plain}
\bibliography{bibliographie}

\end{document}

```



Copier ce code
Code web : 182918

```

@misc
{theo,
author={Wikipédia},
title={Théorème de Pythagore},
month={jul},
year={2010},
note={fr.wikipedia.org}
}

@misc
{mult,
author={Wikipédia},
title={Table de multiplication},
month={jul},
year={2010},
note={fr.wikipedia.org}
}

```

```

}

@misc
{addi,
author={Wikipédia},
title={Table d'addition},
month={jul},
year={2010},
note={fr.wikipedia.org}
}

```

▷ Copier ce code
Code web : 524202

Être plus à l'aise avec L^AT_EX

Ce second TP est un résumé des notions développées dans cet ouvrage. Mine de rien, vous savez déjà beaucoup de choses sur L^AT_EX si vous en êtes à cette page, et ce n'est pas fini ! J'ai rassemblé ici quelques nouvelles commandes, des notions et des outils qui ne vont pas révolutionner votre utilisation de L^AT_EX, mais la rendre plus facile et plus intuitive.

Écrire un document dans plusieurs fichiers .tex

Le concept

Le livre que vous tenez entre vos mains est le fruit d'une compilation d'un fichier nommé `latex.tex`, je vous laisse imaginer la longueur du code nécessaire à la création d'un petit pavé de ce type... Ce sera long, très long même. Fort heureusement, L^AT_EX sait gérer les longs documents et aide les auteurs à scinder une même publication en plusieurs parties⁵. L'utilité de la chose ? Permettre à chacun de travailler sur une partie sans toucher au reste. Ainsi, au moment où j'écris ces lignes, des correcteurs sont en train de retoucher un autre chapitre et nous ne nous dérangeons pas mutuellement⁶. La commande sur laquelle repose ce principe est toute simple, c'est la suivante :

```
| \input{chemin_de_mon_fichier}
```

Lorsque vous utilisez `\input`, L^AT_EX remplace simplement la commande par le contenu du fichier appelé. Un exemple ne ferait pas de mal, n'est-ce pas ?

5. Dans le cadre de ce livre, chaque chapitre fait l'objet d'un fichier différent, mais j'aurais très bien pu choisir de faire autrement.

6. Nous utilisons en plus de cela un logiciel qui nous permet de vérifier que nous n'écrasons pas le travail des autres, au cas où deux personnes travailleraient sur un même fichier.

Exemple

Création des différents fichiers Nous allons créer quatre fichiers, le premier s’appelle `livre.tex` et contiendra le code ci-dessous :

```
\documentclass{book}

\begin{document}

\input{chapitre1.tex}

\input{chapitre2.tex}

\imput{partie2/chapitre3.tex}

\end{document}
```

Les trois autres fichiers (`chapitre1.tex`, `chapitre2.tex` et `chapitre3.tex` — ce dernier étant enregistré dans un dossier nommé `partie 2`) contiennent les codes ci-dessous.

`chapitre1.tex`

```
\chapter{Le premier}
```

`chapitre2.tex`

```
\chapter{Le second}
Poulpy est une déesse.
```

`chapitre3.tex`

```
\chapter{Le dernier}
```

Normalement, l’arborescence de vos fichiers doit être la suivante :

- dans un dossier quelconque, votre fichier `livre.tex`, ainsi que `chapitre1.tex`, `chapitre2.tex` et un dossier nommé `partie2`;
- dans le dossier `partie2`, le fichier `chapitre3.tex`.

Le résultat

La commande `\input` va faire son travail et lors de la compilation, elle va être remplacée par le contenu du fichier vers lequel elle pointe⁷. Au cours de la compilation, votre fichier va ressembler à ça :

7. Tout ceci n’est pas visible, c’est la petite cuisine interne du compilateur.

```

\documentclass{book}

\begin{document}

% Début du contenu de chapitre1.tex
\chapter{Le premier}
% Fin du contenu de chapitre1.tex
% Début du contenu de chapitre2.tex
\chapter{Le second}
Poulpy est une déesse.
% Fin du contenu de chapitre2.tex
% Début du contenu de chapitre3.tex
\chapter{Le dernier}
% Fin du contenu de chapitre3.tex

\end{document}

```

Ce n'est pas plus compliqué que ça. Vous savez maintenant tout ce qu'il faut savoir sur la commande `\input`, vous pouvez donc l'utiliser pour dissocier les différents morceaux de vos publications.

L'option `draft`

Il fut un temps où vous étiez jeune et innocent, et où vous ne connaissiez pas les classes de document. Vous avez utilisé à travers les divers exemples les options liées aux tailles de polices par défaut et d'autres petites choses. Il est temps de vous apprendre un nouvel argument pour `\documentclass` : l'argument `draft`. Le mot *draft* est un terme anglais qui signifie « brouillon » ou « ébauche ». Pour nous, cela veut dire deux choses :

- les images sont remplacées par des cadres blancs contenant leur chemin (absolu ou relatif) ;
- les problèmes de dépassement dans les marges peuvent être facilement repérés en vue de les corriger⁸.

La conséquence logique, c'est que les images ne sont pas vraiment chargées et donc, que la compilation est plus rapide. À titre d'exemple, ce livre a compilé en 65 secondes en mode normal et en 50 secondes en `draft`. Je recommande grandement son utilisation en dernière relecture pour vérifier que tous les mots rentrent dans les marges, et son gain en temps de compilation le rend intéressant quand un utilisateur ne travaille pas sur les images, mais qu'il compile souvent. Faites le test sur ce TP, remplacez la première ligne de code par la suivante et constatez le changement.

```

\documentclass[draft]{book}

```

Ici aussi, c'est une option toute simple, mais qui peut changer pas mal de choses et vous apporter énormément de confort.

8. Le `draft` insère de gros carrés noirs là où il y a dépassement, ce qui permet de les repérer extrêmement facilement et de modifier les phrases problématiques.

Les unités de longueur

À de nombreuses reprises, des notions de longueurs ont été abordées, mais sans vraiment les définir. Quelle longueur fait un point par rapport à un centimètre ? Comment faire les conversions ? Pourquoi Poulpy est-elle tellement magnifique ?

Toutes les réponses sont ici !

Trêve de plaisanteries, je vous ai concocté un tableau tout simple comportant le nom des unités de longueur, leur abréviation dans L^AT_EX (par exemple pour 1 centimètre entre crochets, il faut écrire {1cm}) ainsi que leur conversion en millimètres.



Gardez à l’esprit qu’une feuille A4 mesure 210 millimètres de large et 297 de haut. Cela permet souvent de choisir les bonnes longueurs à appliquer aux commandes.

Les unités de longueur		
Nom	Abréviation	Valeur en millimètres
Point	pt	0,35 mm
Millimètre	mm	1,00 mm
Pica	pc	4,21 mm
Cicéro	cc	4,53 mm
Centimètre	cm	10,00 mm
Inch	in	25,4 mm

Les espaces

Le dernier point que nous allons aborder est celui concernant l’espacement hors du mode scientifique. Il se peut que vous ayez besoin à certains moments d’insérer des espaces horizontaux ou verticaux de tailles précises entre deux mots ou deux morceaux d’un document.

Plutôt que de vous donner toute une gamme de commandes aussi indigeste que difficile à mémoriser, je vais seulement vous en donner quatre. Elles prennent toutes en argument une longueur dans l’unité de votre choix, la seule chose qui change étant leur effet ⁹.

- `\hspace{longueur}` insère un espace horizontal de la longueur choisie.
- `\hspace*{longueur}` : idem, sauf que l’espace n’est pas inséré s’il y a un retour à la ligne entre les deux mots concernés par l’espacement.
- `\vspace{hauteur}` insère un espace vertical de la longueur choisie.
- `\vspace*{hauteur}` : ici, l’espace n’est pas inséré s’il y a un saut de page.

9. La combinaison de ces quatre commandes vous permet de faire à peu près n’importe quoi. Comprenez bien qu’il est parfois préférable de modérer l’usage de ces commandes, aussi puissantes que visuellement dangereuses. :-)

En résumé

- Une approche méthodique aide à venir à bout de gros projets L^AT_EX.
- Il est possible, et même recommandé, de séparer un gros document L^AT_EX en plusieurs fichiers. Cela facilite un travail organisé et collaboratif. Pour inclure un fichier `.tex` dans un autre, on utilise la commande `\input`.
- Le mode `draft` aide à compiler plus rapidement un document.
- L^AT_EX fournit des commandes aidant à l'insertion d'espacements de tailles et d'orientations variables : `\hspace` (espace horizontal) et `\vspace` (espace vertical).
- Citer vos sources dans une bibliographie est un gage de sérieux.

Quatrième partie

Annexes

Annexe A

Les caractères spéciaux

Difficulté : 

Les caractères spéciaux sont très nombreux et très utiles lors de la création de documents. J'ai choisi de vous les lister dans cette annexe afin que vous puissiez les avoir sous la main à tout moment.

Les packages utilisés ici ont été les suivants (rien ne change avec l'UTF-8) :

```
\usepackage[latin1]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage[francais]{babel}  
\usepackage{lmodern}  
\usepackage{amsmath}  
\usepackage{amssymb}  
\usepackage{mathrsfs}
```



Environnement mathématique

Lettres grecques

α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>
δ	<code>\delta</code>	ϵ	<code>\epsilon</code>	ζ	<code>\zeta</code>
η	<code>\eta</code>	θ	<code>\theta</code>	ι	<code>\iota</code>
κ	<code>\kappa</code>	λ	<code>\lambda</code>	μ	<code>\mu</code>
ν	<code>\nu</code>	ξ	<code>\xi</code>	π	<code>\pi</code>
ρ	<code>\rho</code>	σ	<code>\sigma</code>	τ	<code>\tau</code>
υ	<code>\upsilon</code>	ϕ	<code>\phi</code>	χ	<code>\chi</code>
ψ	<code>\psi</code>	ω	<code>\omega</code>		
ϑ	<code>\vartheta</code>	ϖ	<code>\varpi</code>	φ	<code>\varphi</code>
ϱ	<code>\varrho</code>	ε	<code>\varepsilon</code>	ς	<code>\varsigma</code>
Γ	<code>\Gamma</code>	Λ	<code>\Lambda</code>	Σ	<code>\Sigma</code>
Ψ	<code>\Psi</code>	Δ	<code>\Delta</code>	Ξ	<code>\Xi</code>
Υ	<code>\Upsilon</code>	Ω	<code>\Omega</code>	Θ	<code>\Theta</code>
Π	<code>\Pi</code>	Φ	<code>\Phi</code>		

Opérateurs extensibles

$\sum$	<code>\sum</code>	$\int$	<code>\int</code>	$\biguplus$	<code>\biguplus</code>
$\bigoplus$	<code>\bigoplus</code>	$\bigvee$	<code>\bigvee</code>	$\prod$	<code>\prod</code>
$\oint$	<code>\oint</code>	$\bigcap$	<code>\bigcap</code>	$\bigotimes$	<code>\bigotimes</code>
$\bigwedge$	<code>\bigwedge</code>	$\coprod$	<code>\coprod</code>	$\iint$	<code>\iint</code>
$\iiint$	<code>\iiint</code>	$\bigodot$	<code>\bigodot</code>	$\bigsqcup$	<code>\bigsqcup</code>
$\bigcup$	<code>\bigcup</code>				

Fonctions

$\arccos$	<code>\arccos</code>	$\cos$	<code>\cos</code>	$\csc$	<code>\csc</code>
$\exp$	<code>\exp</code>	$\ker$	<code>\ker</code>	$\limsup$	<code>\limsup</code>
$\min$	<code>\min</code>	$\sinh$	<code>\sinh</code>	$\arcsin$	<code>\arcsin</code>
$\cosh$	<code>\cosh</code>	$\deg$	<code>\deg</code>	$\gcd$	<code>\gcd</code>
$\lg$	<code>\lg</code>	$\ln$	<code>\ln</code>	$\Pr$	<code>\Pr</code>
$\sup$	<code>\sup</code>	$\arctan$	<code>\arctan</code>	$\cot$	<code>\cot</code>
$\det$	<code>\det</code>	$\hom$	<code>\hom</code>	$\lim$	<code>\lim</code>
$\log$	<code>\log</code>	$\sec$	<code>\sec</code>	$\tan$	<code>\tan</code>
$\arg$	<code>\arg</code>	$\coth$	<code>\coth</code>	$\dim$	<code>\dim</code>
$\inf$	<code>\inf</code>	$\liminf$	<code>\liminf</code>	$\max$	<code>\max</code>
$\sin$	<code>\sin</code>	$\tanh$	<code>\tanh</code>		

Flèches

$\leftarrow$	<code>\leftarrow</code>	$\longleftarrow$	<code>\longleftarrow</code>	$\uparrow$	<code>\uparrow</code>
$\Leftarrow$	<code>\Leftarrow</code>	$\Longleftarrow$	<code>\Longleftarrow</code>	$\Uparrow$	<code>\Uparrow</code>
$\rightarrow$	<code>\rightarrow</code>	$\longrightarrow$	<code>\longrightarrow</code>	$\downarrow$	<code>\downarrow</code>
$\Rightarrow$	<code>\Rightarrow</code>	$\Longrightarrow$	<code>\Longrightarrow</code>	$\Downarrow$	<code>\Downarrow</code>
$\leftrightarrow$	<code>\leftrightarrow</code>	$\longleftrightarrow$	<code>\longleftrightarrow</code>	$\updownarrow$	<code>\updownarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Longleftrightarrow$	<code>\Longleftrightarrow</code>	$\Updownarrow$	<code>\Updownarrow</code>
$\mapsto$	<code>\mapsto</code>	$\longmapsto$	<code>\longmapsto</code>	$\nearrow$	<code>\nearrow</code>
$\hookleftarrow$	<code>\hookleftarrow</code>	$\hookrightarrow$	<code>\hookrightarrow</code>	$\searrow$	<code>\searrow</code>
$\leftharpoonup$	<code>\leftharpoonup</code>	$\rightharpoonup$	<code>\rightharpoonup</code>	$\swarrow$	<code>\swarrow</code>
$\leftharpoondown$	<code>\leftharpoondown</code>	$\rightharpoondown$	<code>\rightharpoondown</code>	$\nwarrow$	<code>\nwarrow</code>
$\rightleftharpoons$	<code>\rightleftharpoons</code>	$\leadsto$	<code>\leadsto</code>	$\dashrightarrow$	<code>\dashrightarrow</code>
$\dashleftarrow$	<code>\dashleftarrow</code>	$\leftleftarrows$	<code>\leftleftarrows</code>	$\leftrightarrows$	<code>\leftrightarrows</code>
$\Lleftarrow$	<code>\Lleftarrow</code>	$\twoheadleftarrow$	<code>\twoheadleftarrow</code>	$\leftarrowtail$	<code>\leftarrowtail</code>
$\looparrowleft$	<code>\looparrowleft</code>	$\leftrightharpoons$	<code>\leftrightharpoons</code>	$\curvearrowleft$	<code>\curvearrowleft</code>
$\circlearrowleft$	<code>\circlearrowleft</code>	$\Lsh$	<code>\Lsh</code>	$\upuparrows$	<code>\upuparrows</code>
$\upharpoonleft$	<code>\upharpoonleft</code>	$\downharpoonleft$	<code>\downharpoonleft</code>	$\multimap$	<code>\multimap</code>
$\leftrightsquigarrow$	<code>\leftrightsquigarrow</code>	$\rightrightarrows$	<code>\rightrightarrows</code>	$\rightleftarrows$	<code>\rightleftarrows</code>
$\rightrightarrows$	<code>\rightrightarrows</code>	$\rightleftarrows$	<code>\rightleftarrows</code>	$\twoheadrightarrow$	<code>\twoheadrightarrow</code>
$\rightarrowtail$	<code>\rightarrowtail</code>	$\looparrowright$	<code>\looparrowright</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>
$\curvearrowright$	<code>\curvearrowright</code>	$\circlearrowright$	<code>\circlearrowright</code>	$\Rsh$	<code>\Rsh</code>
$\downdownarrows$	<code>\downdownarrows</code>	$\upharpoonright$	<code>\upharpoonright</code>	$\downharpoonright$	<code>\downharpoonright</code>
$\rightsquigarrow$	<code>\rightsquigarrow</code>	$\nleftarrow$	<code>\nleftarrow</code>	$\nrightarrow$	<code>\nrightarrow</code>
$\nLeftarrow$	<code>\nLeftarrow</code>	$\nrightarrow$	<code>\nrightarrow</code>	$\nleftrightarrow$	<code>\nleftrightarrow</code>
$\nLeftrightarrow$	<code>\nLeftrightarrow</code>				

Opérateurs

$*$	<code>\ast</code>	$\pm$	<code>\pm</code>	$\cap$	<code>\cap</code>
$\triangleleft$	<code>\lhd</code>	$\star$	<code>\star</code>	$\mp$	<code>\mp</code>
$\cup$	<code>\cup</code>	$\triangleright$	<code>\rhd</code>	$\cdot$	<code>\cdot</code>
$\amalg$	<code>\amalg</code>	$\oplus$	<code>\oplus</code>	$\triangleleft$	<code>\triangleleft</code>
$\circ$	<code>\circ</code>	$\odot$	<code>\odot</code>	$\sqcap$	<code>\sqcap</code>
$\triangleright$	<code>\triangleright</code>	$\bullet$	<code>\bullet</code>	$\ominus$	<code>\ominus</code>
$\sqcup$	<code>\sqcup</code>	$\triangleleft$	<code>\triangleleft</code>	$\bigcirc$	<code>\bigcirc</code>
$\oplus$	<code>\oplus</code>	$\wedge$	<code>\wedge</code>	$\unrhd$	<code>\unrhd</code>
$\diamond$	<code>\diamond</code>	$\oslash$	<code>\oslash</code>	$\vee$	<code>\vee</code>
$\bigtriangledown$	<code>\bigtriangledown</code>	$\times$	<code>\times</code>	$\otimes$	<code>\otimes</code>

†	\dagger	△	\bigtriangleup	÷	\div
ℓ	\wr	‡	\ddagger	\	\setminus
•	\centerdot	□	\Box	∧	\barwedge
∨	\veebar	⊗	\circledast	⊞	\boxplus
⋈	\curlywedge	Υ	\curlyvee	⊙	\circledcirc
⊖	\boxminus	⊎	\Cap	⊃	\Cup
⊖	\circleddash	⊠	\boxtimes	⊤	\top
⊥	\bot	⊕	\dotplus	⊠	\boxdot
⊤	\intercal	×	\rightthreetimes	*	\divideontimes
□	\square	∧	\doublebarwedge	×	\leftthreetimes
≡	\equiv	≤	\leq	≥	\geq
⊥	\perp	ℝ	\cong	⋈	\prec
Υ	\succ	—	\mid	≠	\neq
⋈	\preceq	Υ	\succeq		\parallel
~	\sim	≪	\ll	≫	\gg
⊗	\bowtie	ℝ	\simeq	⊂	\subset
⊃	\supset	⋈	\Join	≈	\approx
⊆	\subseteq	×	\ltimes	⋈	\asymp
⊆	\sqsubset	⊆	\sqsupset	×	\rtimes
⊆	\doteq	⊆	\sqsubseteq	⊆	\sqsupseteq
(	\smile	∝	\propto	⊥	\dashv
⊤	\vdash	)	\frown	⊥	\models
∈	\in	∪	\ni	∉	\notin
ℝ	\approx	≡	\leq	≡	\geq
≠	\lessgtr	~	\thicksim	≡	\leqslant
≠	\geqslant	≡	\lesseqgtr	≡	\backsim
≡	\lessapprox	≡	\gtrapprox	≡	\lesseqqgtr
≡	\backsimeq	≡	\lll	≡	\ggg
≡	\gtreqless	≡	\triangleq	≡	\lessdot
≡	\gtrdot	≡	\gtreqless	≡	\circeq
≡	\lesssim	≡	\gtrsim	≡	\gtrless
≡	\bumpeq	≡	\eqslantless	≡	\eqslantgtr
≡	\backepsilon	≡	\Bumpeq	≡	\prec
≡	\succsim	≡	\between	≡	\doteqdot
≡	\precapprox	≡	\succapprox	≡	\pitchfork
≡	\thickapprox	≡	\Subset	≡	\Supset
—	\shortmid	≡	\fallingdotseq	≡	\subseteq
⊆	\supseteq	)	\smallfrown	≡	\risingdotseq
⊆	\sqsubset	⊆	\sqsupset	(	\smallsmile
∝	\varpropto	⋈	\preccurlyeq	⋈	\succcurlyeq
⊥	\Vdash	∴	\therefore	⋈	\curlyeqprec

$\succcurlyeq$	<code>\curlyeqsucc</code>	$\vDash$	<code>\vDash</code>	$\because$	<code>\because</code>
$\blacktriangleleft$	<code>\blacktriangleleft</code>	$\blacktriangleright$	<code>\blacktriangleright</code>	$\Vdash$	<code>\Vdash</code>
$\equiv$	<code>\eqcirc</code>	$\trianglelefteq$	<code>\trianglelefteq</code>	$\trianglerighteq$	<code>\trianglerighteq</code>
$\parallel$	<code>\shortparallel</code>	$\neq$	<code>\neq</code>	$\vartriangleleft$	<code>\vartriangleleft</code>
$\triangleright$	<code>\vartriangleright</code>	$\nshortparallel$	<code>\nshortparallel</code>	$\ncong$	<code>\ncong</code>
$\ngeq$	<code>\ngeq</code>	$\nsubseteq$	<code>\nsubseteq</code>	$\nmid$	<code>\nmid</code>
$\nleqq$	<code>\nleqq</code>	$\nsupseteq$	<code>\nsupseteq</code>	$\nparallel$	<code>\nparallel</code>
$\nleqslant$	<code>\nleqslant</code>	$\ngeqslant$	<code>\ngeqslant</code>	$\nsubseteqq$	<code>\nsubseteqq</code>
$\nshortmid$	<code>\nshortmid</code>	$\nless$	<code>\nless</code>	$\ngtr$	<code>\ngtr</code>
$\nsupseteqq$	<code>\nsupseteqq</code>	$\nshortparallel$	<code>\nshortparallel</code>	$\nprec$	<code>\nprec</code>
$\nsucc$	<code>\nsucc</code>	$\subsetneq$	<code>\subsetneq</code>	$\nsim$	<code>\nsim</code>
$\npreceq$	<code>\npreceq</code>	$\nsucceq$	<code>\nsucceq</code>	$\supsetneq$	<code>\supsetneq</code>
$\nVDash$	<code>\nVDash</code>	$\precnapprox$	<code>\precnapprox</code>	$\succnapprox$	<code>\succnapprox</code>
$\subsetneqq$	<code>\subsetneqq</code>	$\nvDash$	<code>\nvDash</code>	$\precnsim$	<code>\precnsim</code>
$\succnsim$	<code>\succnsim</code>	$\supsetneqq$	<code>\supsetneqq</code>	$\nvdash$	<code>\nvdash</code>
$\lnapprox$	<code>\lnapprox</code>	$\gnapprox$	<code>\gnapprox</code>	$\varsubsetneq$	<code>\varsubsetneq</code>
$\ntriangleleft$	<code>\ntriangleleft</code>	$\lneq$	<code>\lneq</code>	$\gneq$	<code>\gneq</code>
$\varsupsetneq$	<code>\varsupsetneq</code>	$\ntrianglelefteq$	<code>\ntrianglelefteq</code>	$\lneqq$	<code>\lneqq</code>
$\gneqq$	<code>\gneqq</code>	$\varsubsetneqq$	<code>\varsubsetneqq</code>	$\ntriangleright$	<code>\ntriangleright</code>
$\lnsim$	<code>\lnsim</code>	$\gnsim$	<code>\gnsim</code>	$\varsupsetneqq$	<code>\varsupsetneqq</code>
$\ntrianglerighteq$	<code>\ntrianglerighteq</code>	$\lvertneqq$	<code>\lvertneqq</code>	$\gvertneqq$	<code>\gvertneqq</code>

Accents

$\acute{a}$	<code>\acute{a}</code>	$\bar{a}$	<code>\bar{a}</code>	$\acute{A}$	<code>\Acute{\Acute{a}}</code>
$\bar{a}$	<code>\Bar{\Bar{a}}</code>	$\breve{a}$	<code>\breve{a}</code>	$\check{a}$	<code>\check{a}</code>
$\breve{a}$	<code>\Breve{\Breve{a}}</code>	$\check{a}$	<code>\Check{\Check{a}}</code>	$\ddot{a}$	<code>\ddot{a}</code>
$\dot{a}$	<code>\dot{a}</code>	$\ddot{a}$	<code>\Ddot{\Ddot{a}}</code>	$\dot{A}$	<code>\Dot{\Dot{a}}</code>
$\grave{a}$	<code>\grave{a}</code>	$\hat{a}$	<code>\hat{a}</code>	$\grave{A}$	<code>\Grave{\Grave{a}}</code>
$\hat{a}$	<code>\Hat{\Hat{a}}</code>	$\tilde{a}$	<code>\tilde{a}</code>	$\vec{a}$	<code>\vec{a}</code>
$\tilde{a}$	<code>\Tilde{\Tilde{a}}</code>	$\vec{a}$	<code>\Vec{\Vec{a}}</code>		

Polices mathématiques

Démonstration

```

\[\mathcal{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}\]
\[\mathbb{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}\]
\[\mathbf{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}\]
\[\mathsf{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}\]
\[\mathbf{A B C D E F G H I J K L M N O P Q R S T U V W X Y Z}\]

```

Rendu

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ⒶⒷⒸⒹⒺⒻⒼⒿ⓫⓬⓭⓮⓯⓰⓱⓲⓳⓴⓵⓶⓷⓸⓹⓺⓻⓼⓽⓾⓿

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

Caractères de structure

\$	\\$
&	\&
%	\%
#	\#
{	\{
}	\}
—	_
~	\sim
^	\wedge
\	\backslash

Divers

∞	<code>\infty</code>	$\forall$	<code>\forall</code>	$\mathbb{k}$	<code>\Bbbk</code>
$\wp$	<code>\wp</code>	∇	<code>\nabla</code>	$\exists$	<code>\exists</code>
★	<code>\bigstar</code>	$\angle$	<code>\angle</code>	∂	<code>\partial</code>
$\nexists$	<code>\nexists</code>	$\exists$	<code>\exists</code>	★	<code>\bigstar</code>
$\angle$	<code>\angle</code>	∂	<code>\partial</code>	$\nexists$	<code>\nexists</code>
$\diagdown$	<code>\diagdown</code>	$\measuredangle$	<code>\measuredangle</code>	$\eth$	<code>\eth</code>
$\emptyset$	<code>\emptyset</code>	$\diagup$	<code>\diagup</code>	$\sphericalangle$	<code>\sphericalangle</code>
♣	<code>\clubsuit</code>	$\varnothing$	<code>\varnothing</code>	◇	<code>\Diamond</code>
$\complement$	<code>\complement</code>	◇	<code>\diamondsuit</code>	$\imath$	<code>\imath</code>
$\Finv$	<code>\Finv</code>	∇	<code>\triangledown</code>	♥	<code>\heartsuit</code>
$\jmath$	<code>\jmath</code>	$\oslash$	<code>\Game</code>	△	<code>\triangle</code>
♠	<code>\spadesuit</code>	ℓ	<code>\ell</code>	$\hbar$	<code>\hbar</code>
△	<code>\vartriangle</code>	$\cdots$	<code>\cdots</code>	$\iiint$	<code>\iiiint</code>
$\hslash$	<code>\hslash</code>	◆	<code>\blacklozenge</code>	$\vdots$	<code>\vdots</code>
$\iiint$	<code>\iiint</code>	◇	<code>\lozenge</code>	■	<code>\blacksquare</code>
$\ldots$	<code>\ldots</code>	$\iint$	<code>\iint</code>	U	<code>\mho</code>
▲	<code>\blacktriangle</code>	$\ddots$	<code>\ddots</code>	#	<code>\sharp</code>
$\prime$	<code>\prime</code>	Ⓢ	<code>\circledS</code>	‡	<code>\natural</code>
$\flat$	<code>\flat</code>	□	<code>\square</code>	\	<code>\backprime</code>
$\Re$	<code>\Re</code>	ℑ	<code>\Im</code>	√	<code>\surd</code>
$\times$	<code>\times</code>				

Autres caractères spéciaux



Ces commandes ne s'utilisent pas dans le mode d'écriture scientifique, même si certaines d'entre elles fonctionnent correctement dans un environnement de ce type.

ó	\'{}o}	ö	\"{}o}	ô	\^{}o}
\	\textbackslash{}	õ	\~{}o}	ō	\={}o}
š	\d s	ò	\.{}o}	ö	\u{}o}
õ	\H{o}	ôo	\t{}oo}	œ	\oe
q	\c{o}	o	\d{o}	š	\r s
o	\b{o}	Å	\AA	â	\aa
ß	\ss	ı	\i	ı	\j
š	\H s	ø	\o	š	\t s
š	\v s	Ø	\O	¶	\P
§	\S	æ	\ae	Æ	\AE
†	\dag	‡	\ddag	©	\copyright
£	\pounds	{	\{	}	\}
%	\%	\$	\\$	&	\&
#	\#	—	_		
^	\^{}{}	...	\dots		

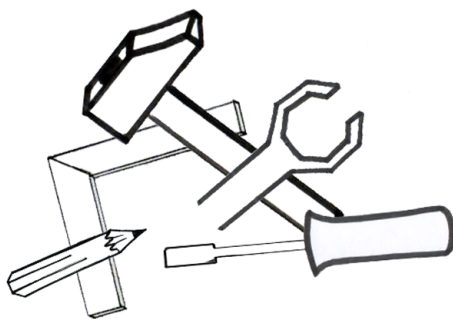
Annexe **B**

Liste des packages

Difficulté : 

Depuis le début du livre, vous avez appris à utiliser quelques packages utiles à la création de documents simples sous $\text{\LaTeX}$. Cette annexe va reprendre, thème par thème, le nom et le rôle de chacun des packages utilisés dans ce cours, avec, pour chacun d'entre eux, un petit commentaire sur son utilité et le besoin auquel il répond.

Garder cette liste sous la main est un bon moyen d'éviter les erreurs d'étourderie — ou tout simplement les fautes de frappe — lors de la saisie de vos préambules. Gardez tout de même à l'esprit qu'un package s'insère via la commande `\usepackage{}`.



La liste

Packages de langue

Ces trois packages sont indispensables pour écrire un document en français.

- `inputenc`
- `fontenc`
- `babel`

Création d'un layout

Le layout permet d'afficher le gabarit de mise en page d'un document.

- `layout`

Modification des marges

Les commandes de ce package permettent de régler finement les marges.

- `geometry`

Interligne

- `setspace`

Soulignement

Le package `soul` peut également servir à barrer du texte.

- `soul`
- `ulem`

Symbole euro

- `eurosym`

Pack de polices

N'en inclure qu'un à la fois (il est fortement conseillé d'en utiliser un).

- `bookman`
- `charter`
- `newcent`
- `lmodern`
- `mathpazo`
- `mathptmx`

Citation d'URL

- `url`

Citation de code

Le package `verbatim` donne accès aux environnements `verbatim` et `verbatimtab`.

- `verbatim`
- `moreverb`

Citation de code coloré

Ne pas oublier de paramétrer correctement `\lstset`.

- `listings`

En-têtes et pieds de pages personnalisés

- `fancyhdr`

Insertion d'images

Primordial pour travailler sur des images.

- `graphicx`

Insertion d'une image dans un paragraphe

À utiliser en dernier recours uniquement...

- `wrapfig`

Manipuler les couleurs et colorer du texte

À employer avec prudence et retenue.

- `color`

Colorer du texte contenu dans un tableau

Pour l'impression, il vaut mieux utiliser uniquement des niveaux de gris.

- `colortbl`

Insertion d'expressions scientifiques

Le package `amsthm` n'est utile que si votre document comprend des théorèmes.

- `amsmath`

- amssymb
- mathrsfs
- amsthm

Création d'index

- makeidx

Petite astuce

Voici une petite astuce toute simple pour ne pas avoir à vous souvenir de chacun des noms des vingt-huit packages utilisés ici : il suffit de créer un gros bout de code à insérer en préambule et dont chaque ligne est précédée par un signe %. Il ne reste qu'à enlever le signe pour qu'une ligne ne soit plus considérée comme un commentaire.

N'hésitez pas à utiliser le code ci-dessous dans vos préambules.

```
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}
%\usepackage{layout}
%\usepackage{geometry}
%\usepackage{setspace}
\usepackage{soul}
\usepackage{ulem}
%\usepackage{eurosym}
%\usepackage{bookman}
%\usepackage{charter}
%\usepackage{newcent}
%\usepackage{lmodern}
%\usepackage{mathpazo}
%\usepackage{mathptmx}
%\usepackage{url}
%\usepackage{verbatim}
%\usepackage{moreverb}
%\usepackage{listings}
%\usepackage{fancyhdr}
%\usepackage{wrapfig}
%\usepackage{color}
%\usepackage{colortbl}
\usepackage{amsmath}
\usepackage{amssymb}
\usepackage{mathrsfs}
%\usepackage{asmthm}
%\usepackage{makeidx}
```



Copier ce code
Code web : 898277

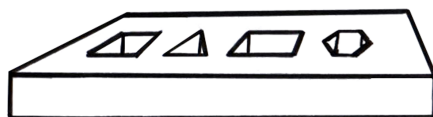
Annexe C

Les gabarits

Difficulté : 

Si vous l'avez lu en entier, ce livre devrait vous avoir convaincu, grâce aux exemples, de la puissance de $\text{\LaTeX}$. Néanmoins, un petit problème se pose : comment retenir les commandes élémentaires ainsi que les éléments de structure à insérer dans un document `.tex` ? Cette annexe rassemble pour vous quatre gabarits de documents afin de faciliter vos compositions de rapports, lettres, livres et articles.

Les différents codes web sont là pour que vous téléchargiez les gabarits sur votre ordinateur et les utilisiez lorsque le besoin s'en fera sentir¹.



1. Ce livre est lui-même créé selon un gabarit : celui des Livres du Zéro, dont la mise en page est parfois fort élaborée. Utiliser les mêmes gabarits pour une série de documents assure une cohérence dans les ouvrages.

La classe article

Notons ici qu'un article scientifique peut parfois faire plus d'une dizaine de pages (c'est même assez courant). Il est souvent primordial de l'accompagner d'une bibliographie, voire d'une table des matières².

```
\documentclass[a4paper,10pt]{article}
\usepackage[utf8x]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[french]{babel}
\usepackage{lmodern} % Pour changer le pack de police
\author{\textsc{Nom} Prénom}
\date{\today}
\title{Mon titre d'article}

\begin{document}

\maketitle

\begin{abstract}
Le résumé (abstract en anglais) de mon article.
\end{abstract}

% Le contenu
Bla bla bla

\end{document}
```



Copier ce code

Code web : 414452

2. Qu'il me semblerait plus judicieux de placer en début de publication dans le cas d'un document de la classe `article`.

La classe report

```
\documentclass[a4paper,10pt]{report}
\usepackage[utf8x]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[french]{babel}
\usepackage{lmodern} % Pour changer le pack de police
\usepackage{makeidx}
\title{Le titre}
\author{\textsc{Nom} Prénom}
\date{} % Pour mettre la date du jour, tapez \today

\title{Mon titre d'article}
\date{}
\author{Laleloulilo}
\makeindex
\begin{document}

\maketitle

\begin{abstract}
Le résumé (abstract en anglais) de mon article.
\end{abstract}

\newpage
\tableofcontents
\newpage

% Le contenu
Bla\index{bla} bla bla

\listoffigures
\listoftables
\printindex
\end{document}
```

▷ Copier ce code
Code web : 428984

La classe book

```
\documentclass{book}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[french]{babel}
\usepackage{lmodern} % Pour changer le pack de police
\usepackage{makeidx}
\makeindex
\title{Le titre}
\author{\textsc{Nom} Prénom}
\date{\today}

\begin{document}

\maketitle % Page de garde

\frontmatter

Pages introductives

\mainmatter

Contenu

\appendix

Chapitres annexes
% Certains mettent la bibliographie après backmatter.
\bibliographystyle{} % Le style est mis entre crochets.
\bibliography{bibli} % Mon fichier de base de données s'appelle bibli.bib.

\backmatter

Epilogue

\tableofcontents
\listoffigures
\listoftables
\printindex

\end{document}
```



Copier ce code
Code web : 886113

La classe letter

Nous avons peu parlé de la classe `letter`, celle-ci peut pourtant s'avérer très utile lors de la rédaction de courriers officiels (ou plus généralement de courriers un peu sérieux). Certains lui préféreront la classe `lettre`, que je trouve un peu moins élégante.

```
\documentclass{letter}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[français]{babel}
\usepackage{lmodern}

\signature{M. Laleloulo (signature)}
\address{Première ligne de l'adresse \\ Deuxième ligne \\ Troisième ligne}
\begin{document}
% Pas besoin d'insérer un champ de datation. Il est ajouté par défaut.
\begin{letter}{Un destinataire\\Un autre \\ Son copain \& son lapin \\
ligne d'adresse 1 \\ ligne d'adresse 2 \\ ligne d'adresse 3}
\opening{Formule de politesse d'ouverture,}

Le texte.

\closing{Cordialement, (formule de politesse)}
\ps{P.-S. : Votre petit ajout ! :-)}

\end{letter}

\end{document}
```

▷ Copier ce code
Code web : 158135

Annexe D

Mémento

Difficulté : 

N ombreux furent les environnements et commandes étudiés tout au long de ce livre. $\text{\LaTeX}$ en compte au moins cent, voire mille fois plus et pourtant, vous avez déjà du mal à vous souvenir de celles qui ont été vues. De plus, vous n'allez pas vous amuser à relire et à feuilleter les 200 pages d'explications pour retrouver une commande ou une syntaxe.

Pour cette raison, j'ai écrit pour vous cette annexe « Mémento » qui rassemble la majeure partie des commandes vues précédemment ainsi que quelques bonus. Certains éléments ne peuvent que difficilement être résumés sans perdre en précision, dans ce cas un renvoi est fait vers une page de cours.

Les structures des différentes classes de document ne sont pas rappelées ici, elles font l'objet de l'annexe « Gabarits ».



Les essentiels

Squelette minimal

Dans le cas d’une publication en français.

```
\documentclass{report}
\usepackage[utf8x]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[francais]{babel}

\begin{document}
\end{document}
```

Appel d’un package

```
\usepackage{nom du package}
```

Introduction d’un environnement

```
\begin{nom de l’environnement}
\end{nom de l’environnement}
```

Inclusion d’un fichier

```
\input{chemin du fichier}
```

La commande \documentclass

Options applicables à la commande \documentclass		
Description	Valeurs applicables	Valeur par défaut
Format du papier	a4paper, a5paper, letterpaper, b5paper...	letterpaper
Mode brouillon	draft	Pas en draft par défaut
Taille de la police principale	10pt, 11pt, 12pt	10pt
Alignement des équations	fleqn (à gauche)	centrées par défaut
Colonnes	onecolumn, twocolumn	onecolumn
Première page des chapitres	openany, openright	openright
Recto verso	oneside, twoside	article et report : oneside, book : twoside

Les références

Les marqueurs sont des éléments invisibles permettant de créer des références¹.

```
\label{historique} % Placer un marqueur nommé "historique".
\ref{historique} % Faire une référence dans un texte.
\pageref{historique} % Faire une référence à la page du marqueur.
```

Les éléments de mise en page

Éléments de structure

Commandes des éléments de structure	
Partie	<code>\part{nom de la partie}</code>
Chapitre	<code>\chapter{nom du chapitre}</code>
Section	<code>\section{nom de la section}</code>
Sous-section	<code>\subsection{nom de la sous-section}</code>
Sous-sous-section	<code>\subsubsection{nom de la sous-sous-section}</code>
Paragraphe	<code>\paragraph{nom du paragraphe}</code>
Sous-paragraphe	<code>\subparagraph{nom du sous paragraphe}</code>

Page de garde

```
\title{Le titre}
\author{Prénom \bsc{Nom}}
\date{\today}
\begin{document}
\maketitle
```

Alignements de texte

Par défaut le texte est justifié. Vous pouvez utiliser :

- l’environnement `flushright` pour aligner votre texte à droite;
- l’environnement `center` pour centrer votre texte;
- l’environnement `flushleft` pour aligner votre texte à gauche.

Sauts

```
\newline % Changement de ligne.
\\ % Changement de ligne.
```

1. Placez d’abord un marqueur avec `\label` puis faites-y référence ailleurs dans le texte.

```
\newpage % Changement de page.  
\clearpage % Changement de page après gestion des flottants.  
\cleardoublepage % Idem, mais redémarre sur une page impaire.
```

Interlignes

Pour un interligne 1,5 fois plus grand que celui par défaut, utilisez `\onehalfspacing` dans le préambule ou bien l’environnement `onehalfspace`.

Pour un interligne 2 fois plus grand que celui par défaut, utilisez `\doublespacing` dans le préambule ou bien l’environnement `doublespace`.

Marges

Modification grossière des marges

Exemple avec 3 cm de marge partout :

```
\usepackage[top=3cm, bottom=3cm, left=3cm, right=3cm]{geometry}
```

Création d’un layout

```
\documentclass{article}  
  
\usepackage[latin1]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage[français]{babel}  
\usepackage{layout}  
  
\begin{document}  
  
\layout  
  
\end{document}
```

Modifications fines d’un layout

Les noms des longueurs sont disponibles sur le layout.

```
\setlength{nom de la longueur}{longueur dans l’unité qui vous plait}  
\setlength{\marginparwidth}{2cm}
```

Listes

Listes à puces

```
\begin{document}

\begin{itemize}

\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item[@] Une pintade. % En plaçant un @ entre crochets après \item,
↪ j'ai transformé la puce en @.

\end{itemize}
```

Listes numérotées

```
\begin{enumerate}

\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item Un canard.
\item Un mammouth.
\item Une pintade.

\end{enumerate}
```

Listes de description

```
\begin{description}

\item[Un canard :] bestiole qui fait coin.
\item[Un poulpe :] bestiole qui fait bloub.
\item[Un ornithorynque :] bestiole qui fait rire.
\item[Un ours :] bestiole qui fait mal.

\end{description}
```

Les styles

```
|\pagestyle{nom du style}
```

- le style `plain` : numéro de page au milieu du pied de page.
- le style `headings` : nom du chapitre et numéro de page dans l'en-tête, pied de page vide.
- le style `empty` : en-tête *et* pied de page vides.

En-têtes et pieds de page personnalisés

L'essentiel

`| \pagestyle{fancy}`

- `\lhead{...}` gère le coin gauche de l'en-tête.
- `\chead{...}` gère le centre de l'en-tête.
- `\rhead{...}` gère le coin droit de l'en-tête.
- `\lfoot{...}` gère le coin gauche du pied de page.
- `\cfoot{...}` gère le centre du pied de page.
- `\rlfoot{...}` gère le coin droit du pied de page.

Commandes spécifiques

- `\leftmark` : nom du chapitre.
- `\rightmark` : nom de la section.
- `\thepage` : numéro de page.
- `\thechapter` : numéro du chapitre.
- `\thesection` : numéro de la section courante.
- `\renewcommand{\headrulewidth}{épaisseur dans l'unité de votre choix}` : épaisseur de la ligne entre l'en-tête et le corps du texte.
- `\renewcommand{\footrulewidth}{épaisseur dans l'unité de votre choix}` : épaisseur de la ligne entre le pied de page et le corps du texte.
- `\renewcommand{\headsep}{longueur dans l'unité de votre choix}` : espacement entre l'en-tête et le corps du document.

Changement de nom d'éléments

Syntaxe requise : `\renewcommand{\élément}{Nouveau nom}`

Unités de longueur

Les espaces adaptables

- `\hspace{longueur}` : espace horizontal.
- `\hspace*{longueur}` : espace non inséré en cas de retour à la ligne.
- `\vspace{hauteur}` : espace vertical.
- `\vspace*{hauteur}` : espace non inséré en cas de saut de page.

Texte par défaut de chaque commande	
Commande	Texte en français
<code>\abstractname</code>	Résumé
<code>\appendixname</code>	Annexe
<code>\chaptername</code>	Chapitre
<code>\figurename</code>	Figure
<code>\tablename</code>	Table
<code>\indexname</code>	Index
<code>\listfigurename</code>	Table des figures
<code>\listtablename</code>	Liste des tableaux
<code>\pagename</code>	Page
<code>\partname</code>	Partie
<code>\seename</code>	Voir

Les unités de longueur		
Nom	Abréviation	Valeur en millimètres
Point	pt	0,35 mm
Millimètre	mm	1,00 mm
Pica	pc	4,21 mm
Cicéro	cc	4,53 mm
Centimètre	cm	10,00 mm
Inch	in	25,4 mm

Les polices

Les tailles

Commande	Taille de texte
<code>\tiny</code>	Minuscule
<code>\scriptsize</code>	Très très petite
<code>\footnotesize</code>	Très petite
<code>\small</code>	Petite
<code>\normalsize</code>	Normale (définie dans <code>\documentclass</code>)
<code>\large</code>	Légèrement plus grande que la normale
<code>\Large</code>	Grande
<code>\LARGE</code>	Très grande
<code>\huge</code>	Très très grande
<code>\Huge</code>	Énorme!

Les couleurs

```
| \textcolor{couleur}{texte en couleur}
```

Création de couleur en niveaux de gris :

```
\definecolor{le nom choisi sans espace}{gray}{le nombre à deux décimales}
```

Création de couleur via un mélange de rouge, de vert et de bleu :

```
\definecolor{nom}{rgb}{taux de rouge entre 0 et 1,taux de  
vert,taux de bleu}
```

Changement de police ponctuel

```
{\fontfamily{code_de_la_police}\selectfont mon bout de texte}
```

Code de la police	Exemple
bch	Charter
cmr	Computer Modern
lmr	Latin Modern Roman
lmss	Latin Modern Sans Empattement
lmssq	Latin Modern Sans Emp. Exp.
lmtt	Latin Modern Typewriter
pag	Avant Garde
pcr	Courier
pbk	Bookman
phv	Helvetica
pnc	New Century Schoolbook
ppl	Palatino
ptm	Times
pzc	<i>Zapf Chancery</i>

Les packs de polices

- bookman
- charter
- newcent
- lmodern
- mathpazo
- mathptmx

Mise en forme		
Modification	Commande	Rendu
Normal	<code>{\normalfont un lapin}</code> <code>\begin{rm}un lapin \end{rm}</code>	un lapin un lapin
Mise en avant	<code>\emph{un lapin}</code>	
Gras	<code>\textbf{un lapin}</code> <code>{\bfseries un lapin}</code> <code>\begin{bf}un lapin \end{bf}</code>	un lapin un lapin un lapin
Italique	<code>\textit{un lapin}</code> <code>{\itshape un lapin}</code> <code>\begin{it}un lapin \end{it}</code>	<i>un lapin</i> <i>un lapin</i> <i>un lapin</i>
Penché	<code>\textsl{un lapin}</code> <code>{\slshape un lapin}</code> <code>\begin{sl}un lapin \end{sl}</code>	<i>un lapin</i> <i>un lapin</i> <i>un lapin</i>
Machine à écrire	<code>\texttt{un lapin}</code> <code>{\ttfamily un lapin}</code> <code>\begin{tt}un lapin \end{tt}</code>	un lapin un lapin un lapin
Petites capitales	<code>\textsc{un lapin}</code> <code>\bsc{un lapin}</code> <code>{\scshape un lapin}</code> <code>\begin{sc}un lapin \end{sc}</code>	UN LAPIN UN LAPIN UN LAPIN UN LAPIN
Exposant (En mode texte)	<code>Un canardun lapin</code>	Un canard ^{un lapin}
Encadrer (Paramétrable)	<code>\fbox{un lapin}</code>	un lapin
Soulignement (Package soul)	<code>\ul{un lapin}</code>	<u>un lapin</u>
Soulignement (Package ulem)	<code>\uuline{un lapin}</code>	<u>un lapin</u>
Soulignement (Package ulem)	<code>\uwave{un lapin}</code>	<u>un lapin</u>
Barrer (Package soul)	<code>\st{un lapin}</code>	un lapin

Les effets

Les notes

Citations simples

L'environnement `quote` pour des citations courtes et `quotation` pour des citations de plusieurs lignes.

Citations d'URL :

```
| \url{adresse}
```

Citations de code : voir page 87.

La commande `\fbox`

```
| % Commande permettant de définir l'écart.  
| \setlength{\fboxsep}{8mm}  
| % Commande permettant de définir l'épaisseur du trait.  
| \setlength{\fboxrule}{2mm}  
| \fbox{Un lapin}
```

Création de `minipage`

```
| \begin{minipage}[c]{5cm}  
| Un petit paragraphe.  
| \end{minipage}
```

Les figures

Insertion simple

```
| \includegraphics{chemin de l'image}  
| \includegraphics[width=300]{chemin de l'image}  
| \includegraphics[height=250]{chemin de l'image}  
| \includegraphics[height=250, width=300]{chemin de l'image}  
| \includegraphics[scale=1.2]{chemin de l'image}
```

Rotation

```
| \includegraphics[angle=52]{chemin de l'image}
```

Inclusion dans un paragraphe

```
\begin{wrapfigure}[nombre de lignes]{placement}{largeur de l'image en cm}
\includegraphics[width=largeur en cm]{chemin de l'image}
\end{wrapfigure}
Votre paragraphe, sans saut de ligne entre \end et le paragraphe.
```

Découpage d'une image

```
\includegraphics*[abscisse du coin inférieur gauche,
↪ ordonnée du coin inférieur gauche][abscisse du coin supérieur droit,
↪ ordonnée du coin supérieur droit du rectangle]{chemin de l'image}
```

Image flottante

```
\begin{figure}
\begin{center}
\includegraphics{chemin de l'image}
\end{center}
\caption{Légende}
\label{Référence}
\end{figure}
```

ou

```
\begin{figure}
\center
\includegraphics{chemin de l'image}
\caption{Légende}
\label{Référence}
\end{figure}
```

Les tableaux

Structure type

```
\begin{table}
\begin{center}
\begin{tabular}{|c|c|}
\hline
1 & 2 \\
\hline
3 & 4 \\
\hline
\end{tabular}
\end{center}
\end{table}
```

```
\end{center}  
\caption{votre légende}  
\label{votre référence}  
\end{table}
```

Fusion de lignes

```
\multirow{nombre de lignes fusionnées}{taille de la colonne en cm}{votre texte}  
\multirow{nombre de lignes fusionnées}*{votre texte}
```

Fusion de colonnes

```
\multicolumn{nbre de col fusionnées}{c, l ou r }{le texte}
```

Utilisation de commandes et d’environnements

Voir page 152.

Coloration de cellules

Utilisable avec les couleurs personnalisées.

- `\columncolor{couleur}` pour colorer les colonnes.
- `\rowcolor{couleur}` pour colorer les lignes.
- `\cellcolor{couleur}` pour colorer les cellules.

Les mathématiques

Les formules simples

Fraction

```
\[\frac{numérateur}{dénominateur}\]
```

Fractions en cascade

```
\[\cfrac{num1}{den1 + \cfrac{num2}{den2}}\]
```

Racine

```
\[\sqrt[15]{nombre}\]
```

Limite

`\[\lim_{nbr1 \to nbr2} x+42\]`

Exposant

`\[nombre^{exposant}\]`

Indice

`\[nombre_{indice}\]`

Intégrales simples

`\[\int {contenu}\]`
`\[\int_{borne inférieure}^{borne supérieure} {contenu}\]`
`\[\int \limits_{borne inférieure}^{borne supérieure} {contenu}\]`

Pour faire des intégrales doubles ou triples, utilisez respectivement les commandes `\iint` et `\iiint`².

Intégrale sur une courbe fermée

`\[\oint {contenu}\]`

Symbole de la somme

`\[\sum_{en dessous}^{au dessus} contenu\]`

Symbole du produit

`\[\prod_{en dessous}^{au dessus} contenu\]`

Les systèmes d'équation

`\begin{align}`
`y &= x - y + z \\`
`x &= y \\`
`z &= y`
`\end{align}`
`\begin{align*} % Sans numérotation.`

2. Vous pouvez mettre plusieurs `\int` à la suite, mais mathématiquement cela ne représente pas la même chose.

```
y &=& x - y + z \\
x &=& y \\
z &=& y
\end{align*}

\begin{align}
y &=& x - y + z \\
x &=& y \\
z &=& y
\end{align}
```

L'environnement array

Voir page 168.

Les matrices

Utilisation classique

```
\[
\begin{matrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{matrix}
\end{matrix}
\]
```

Il est aussi possible d'utiliser les environnements `pmatrix`, `vmatrix`, `Vmatrix`, `bmatrix` et `Bmatrix` selon le type de délimiteur souhaité.

La commande `\phantom`

```
\[
\begin{matrix}
\phantom{9}1 & 12345 & \phantom{-}3 \\
94 & \phantom{1234}5 & -6 \\
\phantom{9}7 & \phantom{1234}8 & \phantom{-}9
\end{matrix}
\end{matrix}
\]
```

Les théorèmes, lemmes, etc.

Dans le préambule :

```
\newtheorem*{petit_nom}{Réciproque}
\newtheorem{petit_nom1}{Définition}
```

```
\newtheorem{petit_nom2}{Proposition}[chapter]
\newtheorem{petit_nom3}{Corollaire}[section]
```

Ensuite chaque nouvel environnement défini est utilisé :

```
\begin{petit_nom}
\end{petit_nom}
```

ou

```
\begin{petit_nom}[de quelque chose]
\end{petit_nom}
```

Les espaces

Espace	Code
négative	\!
fine	\,
normale	\ suivi d'une espace
moyenne	\;
large	\:
cadratin	\quad
double cadratin	\qquad

Les tables

```
\tableofcontents % Table des matières.
\listoffigures % Table des figures.
\listoftables % Liste des tableaux.
```

Index

Il faut insérer la commande `\makeindex` dans le préambule ainsi que `\printindex` à l'endroit où vous souhaitez mettre en page votre index.

```
\index{mot} % ajoute l'entrée « mot »
\index{pate@pâté} % « pâté » correctement placé alphabétiquement
\index{mot|see{lapin}} % « mot, voir lapin »
```

Bibliographie

Base de données

```
@type_d'œuvre{référence_courte,
Titre_du_champ1={texte de votre choix},
Titre_du_champ2={texte de votre choix},
Titre_du_champ3={texte de votre choix},
Titre_du_champ4={texte de votre choix},
}
```

Types d'œuvres		
Type d'œuvre	Correspondance	Champs disponibles
@article	Article	author, journal, title, year, month, pages, notes
@book	Livre	author, title, publisher, year
@manual	Document technique	title, author, year, organization
@misc	Divers	author, title, month, year, note
@phdthesis	Thèse de docotorat	title, author, school, year
@unpublished	Manuscrit non publié	author, title, note, year, month

Champs	
Champs	Correspondance
author	Auteur
journal	Journal
month	Mois de publication, au choix (jan, feb, mar, apr...)
note	Notes complémentaires
organization	Société ou organisation
pages	Pages concernées
publisher	Éditeur
year	Année de publication

Insertion de la bibliographie

```
\bibliographystyle{ } % Le style
\bibliography{fichier sans extension} % Le fichier de base de données.
```

- \bibliographystyle{plain} classe les entrées par ordre alphabétique et les numérote en conséquence.
- \bibliographystyle{abbrv} classe les entrées par ordre alphabétique, les numérote en conséquence et abrège certains éléments de la bibliographie.
- \bibliographystyle{unsrt} trie les entrées par ordre d'apparition dans le texte.
- \bibliographystyle{alpha} : le repère n'est plus un chiffre, mais les trois premières lettres du nom de l'auteur accolées aux deux derniers chiffres de l'année de parution.

Annexe E

Aller plus loin

Difficulté : 

Avant de vous quitter, j'ai souhaité mettre à votre disposition une annexe d'ouverture, une sorte de conclusion vous permettant de poursuivre votre apprentissage de $\text{\LaTeX}$ facilement et ce, sans dépenser un sou de plus.

L'idée ici est de vous orienter vers différentes pistes et de vous présenter des idées et points de vue qui vous permettront de vous perfectionner dans la pratique de $\text{\LaTeX}$.



Typographie et encodage

Latin-1 et UTF-8

Un débat a fait rage à diverses étapes de la composition de ce livre : quel encodage de caractères vous apprendre à utiliser ? UTF-8 ou Latin-1 ? Comment vous transmettre les bonnes pratiques sans pour autant vous rendre la vie impossible ? Telles étaient les questions, je vais ici y répondre.

D'abord, si vous étiez endormi à la lecture du passage concerné, sachez que l'encodage est, grossièrement, le nom donné au processus qui transforme vos caractères en code informatique. Plusieurs processus ont en effet été développés, et le standard actuel est l'UTF-8 car il gère très bien plusieurs dizaines de langues ainsi que des centaines de caractères spéciaux.

Vous en déduirez que le Latin-1 a des lacunes dans ces domaines¹ et commencerez doucement à vous demander pourquoi je l'ai recommandé aux utilisateurs de Windows et de Mac.

La raison est simple : généralement, les utilisateurs de Mac et de Windows échangent des foules de fichiers et Windows a encore beaucoup de mal avec l'UTF-8². La probabilité est donc extrêmement forte que vous ayez à travailler avec des personnes utilisant encore le Latin-1, même si c'est mal.



Je suis sous Windows et une personne me communique un document en UTF-8 à modifier, que faire ?

Tout simplement télécharger la version 2 de $\text{\TeX}$ nicCenter, qui fonctionne très bien, même si elle n'est pas encore totalement stable. Une fois qu'il sera en version finale, ce programme comblera le retard pris par Windows et permettra à tous les utilisateurs de $\text{\LaTeX}$ de travailler en UTF-8. Bien sûr, cela prendra du temps, beaucoup de temps. . .

Si vous voulez le tester, voici le code web vous permettant de le télécharger :

▷ Code web : 409060

L'intégriste est partout

Vous aimez les débats qui tachent, le sang qui coule, et comprendre les luttes mesquines entre pro et anti $\text{\eqarray}$ vous intéresse ? J'ai sélectionné pour vous un document qui liste les méfaits de cet environnement ainsi que de nombreux autres. Il s'agit d'un texte à lire pour se sentir malin dans un cadre scientifique et ne pas se faire avoir en produisant ses premiers écrits sérieux. Ce document explique aussi pourquoi $\text{\$}$ n'est

1. Comme vous avez pu le lire dans le TP 1, il n'y a pas besoin de remplacer le caractère $\text{\oe}$ par une commande en Latin-1.

2. Pour tout vous dire, ce livre a été écrit sous Windows en UTF-8 et j'ai mis deux jours à trouver un logiciel ne déformant pas affreusement mes caractères.

plus beaucoup employé pour écrire des expressions scientifiques.

▷ Code web : 762674

De l'art de la présentation

Continuons dans l'ouverture : je vous propose ici deux manuscrits de typographie qui vous permettront, l'un comme l'autre, de comprendre quelles sont les choses à faire³ pour préserver notre langue et être crédible typographiquement face à des magazines ou un lectorat très rigoureux sur l'écriture et le respect des conventions.

▷ Code web : 206680

▷ Code web : 718104

Les cours d'approfondissement

Voici une petite sélection de divers cours à lire si vous voulez approfondir vos connaissances. À votre niveau actuel, aucun ne présente de difficulté particulière⁴. Dans l'ordre des codes web, je vous ai mis :

1. un exemple de cours sous forme de code commenté ;
▷ Code web : 499093
2. le tutoriel de formation à L^AT_EX de l'ENS (un bon niveau général est requis) ;
▷ Code web : 545014
3. le Framabook « Tout ce que vous avez toujours voulu savoir sur L^AT_EX sans jamais oser le demander » ;
▷ Code web : 880879
4. le Wikibooks en français, très bien construit, traitant de L^AT_EX. Il donne des exemples intéressants d'utilisation du langage en chimie ;
▷ Code web : 845501
5. le Wikibooks en anglais, un peu plus fourni ;
▷ Code web : 251599
6. le cours de Ukonline, très répandu chez les débutants ;
▷ Code web : 382354
7. une courte introduction à L^AT_EX, véritable perle de pédagogie ;
▷ Code web : 490205
8. la version web de ce livre, en constante évolution.
▷ Code web : 660251

3. Et à ne surtout pas faire.

4. L^AT_EX utilise toujours les mêmes notions de commandes, d'environnements et de packages ; il n'y a pas grand-chose à apprendre de plus. Si vous aimez le challenge, vous pouvez tenter de créer vos propres commandes et packages.

Les incontournables

Quatre outils fort utiles :

1. Le conjugueur, véritable Bescherelle informatique⁵ ;
▷ [Code web : 677260](#)
2. un aide-mémoire supplémentaire ;
▷ [Code web : 841327](#)
3. une table listant pratiquement tous les caractères spéciaux ;
▷ [Code web : 463736](#)
4. le guide « Détecter et résoudre les problèmes », outil salvateur lorsqu'une erreur de compilation inconnue fait surface.
▷ [Code web : 712577](#)

Beamer

Si vous avez eu le courage de pratiquer, la classe de documents **beamer** vous tend les bras. Cette classe est conçue pour produire des diapositives et se présente comme un concurrent très sérieux de PowerPoint dans la sphère scientifique.

1. La page **Beamer** sur Wikipedia.
▷ [Code web : 649321](#)
2. Introduction à **Beamer** sous Linux.
▷ [Code web : 190634](#)
3. Manuel complet en anglais.
▷ [Code web : 722689](#)

Les interfaces graphiques alternatives

Si vous êtes lassé de Kile, de T_EXnicCenter ou de T_EXShop, prenez le temps de tester les trois logiciels présentés ci-dessous.

1. L'éditeur L^AT_EX en ligne de Google.
▷ [Code web : 143599](#)
2. Lyx, le presque traitement de texte en L^AT_EX.
▷ [Code web : 951477](#)
3. Emacs, la console par excellence.
▷ [Code web : 997411](#)

Au plaisir de vous lire un jour prochain, car il ne faut pas l'oublier, L^AT_EX n'est qu'un écran pour vos créations.

⁵. La conjugaison, la grammaire et l'orthographe sont au moins aussi importantes que la mise en page. N'oubliez pas de les soigner.

Index

A	
alignement	57
annexe	<u>voir</u> <code>\appendix</code>
<code>\appendix</code>	52
B	
backslash	35
barrer	78
base de données	199
bibliographie	198
style	203
C	
caractère interdit	34
citation	
code	87
simple	86
URL	90
commande	4
compilation	32
conversion	33
couleur	
créer	80
niveaux de gris	80
rvb	80
texte	79
créateur	
Donald Erwin Knuth	10
Leslie Lamport	10
D	
<code>\documentclass</code>	
article	37
book	37
letter	37
report	37
option	62
DVI	29, 32
E	
élément de structure	50
emphase	79
encadrer	78, 90
en-tête	<u>voir</u> style
environnement	36
espace	
mode écriture scientifique	177
mode texte	226
exposant	161
expression mathématique	158
F	
figure	128
flottant	7, 137
fonction usuelle	160
footnote	<u>voir</u> note de bas de page
formule	4
fraction	160
G	
gras	78
I	
image	<u>voir</u> figure
index	191
indice	161
installation	13
Adobe Reader	19
distribution L ^A T _E X	14

- éditeur L^AT_EX.....14
- Ghostscript.....19
- Kile.....22
- lecteur PDF/PostScript.....14
- T_EXLive.....22
- T_EXnicCenter.....20
- T_EXShop.....22
- intégrale.....162
- interface.....5
- interligne.....65
- italique.....78

- J**
- journalisme scientifique.....3

- L**
- langage.....3
 - de description.....4
- layout.....65
- légende.....140
- limite.....161
- lipsum.....50
- liste
 - à puce.....67
 - de description.....68
 - numérotée.....67
- liste des tableaux.....189
- logiciel.....13
 - Adobe Reader.....19
 - Linux.....13
 - Kile.....22
 - T_EXLive.....22
 - Mac.....13
 - T_EXShop.....22
 - Windows.....13
 - Bloc-Notes.....14
 - Ghostscript.....19
 - Notepad++.....14
 - T_EXnicCenter.....14

- M**
- marge
 - paramétrage précis.....118
 - simple.....63
- matrice.....171
- mémoire.....9
- minipage.....91

- mode brouillon (**draft**).....225

- N**
- norme
 - éditoriale.....7
 - typographique.....7
- note.....7
- note de bas de page.....92

- O**
- OpenOffice.....6

- P**
- package.....40
 - babel**.....41
 - babel**
 - commande spécifique.....42
 - fontenc**.....41
 - inputenc**.....41
- page de garde.....54
- PDF.....8
- petites capitales.....78
- pied de page..... voir style
- police
 - changement de pack.....81
 - changement ponctuel.....84
- PostScript.....14
- PowerPoint.....8
- premier document.....27
- produit.....165
- publication.....6

- R**
- référence.....7
- racine.....160
- rapport de stage.....9
- référence interne.....94
- rétrocompatibilité.....8

- S**
- saut.....57
 - de ligne.....57
 - de page.....59, 140
- somme.....164
- soulignement.....78
- style
 - avancé.....122

simple 70
système d'équations 166

T

table des figures 189
table des matières 182
tableau 143
taille de texte 76
T_EX 10

U

unité de longueur 226

W

Word 5
WYSIWYG 6

Dépôt légal : octobre 2010
ISBN : 978-2-9535278-4-1
Code éditeur : 978-2-9535278
Imprimé en France

Achevé d'imprimer le 20 octobre 2010
sur les presses de ALBEDIA Imprimeurs (Aurillac)
Imprimé sur papier PEFC issu de forêts gérées durablement
N° PEFC : 10-31-1446
ALBEDIA Imprimeurs est agréée IMPRIM'VERT



Mentions légales :
Crédit photo 4^e de couverture : Béranger Zyla
Crédit photo Donald Knuth p.10 : Jacob Appelbaum (Creative Commons)
Conception couverture : Fan Jiyong
Illustrations chapitres : Malika Talbi